

MINIMUM BAYES RISK ACOUSTIC MODEL ESTIMATION AND
ADAPTATION

By
Matthew Gibson

SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY
AT
DEPARTMENT OF COMPUTER SCIENCE
UNIVERSITY OF SHEFFIELD
SHEFFIELD, UK
NOVEMBER 2008

© Copyright by Matthew Gibson, 2008

To my parents. Their support is unconditional and inexhaustible.

Table of Contents

Table of Contents	iii
List of Tables	viii
List of Figures	x
Acknowledgements	xiv
Acronyms	xv
Notation	xvi
1 Introduction	1
1.1 Automatic speech recognition	2
1.2 Pattern classification	3
1.2.1 Statistical pattern classification and ASR	3
1.3 Acoustic model estimation	4
1.4 Acoustic model adaptation	5
1.5 Objectives	5
1.6 Overview	6
2 Fundamentals of ASR	10
2.1 Feature extraction	10
2.2 Pattern classification and ASR	12
2.2.1 Bayesian decision theory	12
2.2.2 Bayesian decision theory and ASR	13
2.3 Acoustic modelling	13
2.3.1 Coarticulation and acoustic parameter tying	15
2.4 Language modelling	16
2.5 ASR search algorithm	17
2.6 System evaluation	19

2.7	Summary	19
3	Discriminative acoustic model estimation	20
3.1	Generative learning	20
3.1.1	Bayesian inference	21
3.1.2	Maximum a-posteriori model estimation	21
3.1.3	Maximum likelihood model estimation	21
3.1.4	Problems with generative learning	23
3.2	Discriminative learning	24
3.2.1	Conditional Bayesian inference	25
3.2.2	Empirical risk	26
3.2.3	Minimum classification error	27
3.2.4	Maximum margin estimation	28
3.2.5	Minimum Bayes risk	30
3.3	Summary	32
4	Discriminative acoustic model adaptation	33
4.1	Speaker and environment adaptation	33
4.1.1	Speaker adaptation	33
4.1.2	Environment adaptation	34
4.2	Feature-based adaptation	34
4.3	Model-based adaptation	35
4.3.1	MAP adaptation	35
4.3.2	Maximum likelihood linear regression	36
4.3.3	Speaker Adaptive Training	39
4.3.4	Speaker space adaptation	39
4.3.5	Summary	40
4.4	Discriminative speaker adaptation	40
4.4.1	Previous work	41
4.5	Summary	42
5	MBR theory and implementation	44
5.1	MBR criterion optimisation	44
5.1.1	Discrete approximation of a continuous distribution	45
5.1.2	Weak sense auxiliary function	47
5.1.3	Extended Baum-Welch updates for general functions	48
5.1.4	Auxiliary function for the MBR criterion	48
5.2	Implementation of MBR parameter updates	50
5.2.1	Lattice-based MBR	50
5.2.2	Forward-backward algorithms	51
5.2.3	Learning rate D	52

5.2.4	I-smoothing	54
5.2.5	Acoustic scaling and language model specificity	55
5.3	Minimum Bayes risk linear regression	56
5.3.1	I-smoothing for MBR linear regression	57
5.3.2	Complexity control and MBRLR adaptation	57
5.4	Summary	58
6	MBR error approximation	59
6.1	Levenshtein distance approximation	59
6.1.1	Lattice segmentation	60
6.1.2	Alignment-based error approximation	60
6.2	Limitations of baseline approximate error	63
6.2.1	Error overestimation	63
6.2.2	Asymmetry	63
6.2.3	Insertion to deletion bias	63
6.3	Alternative error approximations	63
6.3.1	Frame error normalisation	64
6.3.2	Using multiple reference alignments	66
6.4	Error approximation analysis	69
6.4.1	Raw error approximation	70
6.4.2	Error approximation accuracy	72
6.5	Evaluation: MBR-estimated acoustic models	74
6.5.1	Unsmoothed MBR	74
6.5.2	I-smoothed MBR	76
6.6	Summary and future work	78
6.6.1	Future work	78
7	Sub-word MBR criteria	79
7.1	Introduction	79
7.2	Motivating phoneme-level MBR	80
7.2.1	Focus on acoustic confusion	80
7.2.2	Additional word-end silence discrimination	83
7.3	Phoneme-sensitive word-level MBR criterion	85
7.4	Silence-sensitive word-level MBR criterion	86
7.5	Model and state-level MBR criteria	87
7.6	Similarity of MBR criteria	89
7.6.1	Discussion	90
7.7	Evaluation: sub-word MBR criteria	90
7.7.1	Unsmoothed MBR	91
7.7.2	I-smoothed MBR	92
7.8	Summary and future work	95

7.8.1	Future work	95
8	Confidence-driven MBR acoustic model adaptation	97
8.1	Introduction	97
8.2	Confidence estimation	98
8.2.1	Posterior-based confidence measures	99
8.2.2	Sub-word confidence measures	100
8.3	Confidence-driven MLLR	102
8.3.1	Confidence-thresholded MLLR	102
8.3.2	Confidence-weighted MLLR	103
8.3.3	Sub-word confidence-driven MLLR	103
8.3.4	Confidence-driven MLLR and complexity control	104
8.4	Confidence-driven MBRLR	105
8.4.1	Confidence-thresholded MBRLR	106
8.4.2	Confidence-weighted MBRLR	107
8.4.3	Sub-word confidence-driven MBRLR	107
8.4.4	Confidence-driven MBRLR and generalisation	108
8.4.5	Confidence-driven I-smoothing for MBRLR	109
8.4.6	Confidence-driven complexity control for MBRLR	109
8.4.7	Summary	110
8.5	Evaluation system	110
8.6	Evaluation: confidence measures	111
8.6.1	Performance evaluation	112
8.6.2	Analysis	113
8.7	Evaluation: confidence-driven MLLR	118
8.7.1	Experiment 1: Confidence-thresholded MLLR	118
8.7.2	Experiment 2: Confidence-weighted MLLR	118
8.7.3	Experiment 3: Ideal confidence-driven MLLR	120
8.7.4	Summary	122
8.8	Evaluation: confidence-driven MBRLR	122
8.8.1	Experiment 1: Standard MBRLR	123
8.8.2	Experiment 2: Confidence-thresholded MBRLR	124
8.8.3	Experiment 3: Sub-word confidence-thresholded MBRLR	125
8.8.4	Experiment 4: Confidence-weighted MBRLR	127
8.8.5	Experiment 5: Ideal confidence-driven MBRLR	128
8.8.6	Experiment 6: Confidence-thresholded MBRLR and I-smoothing	130
8.9	Summary and future work	132
8.9.1	Future work	133

9	Conclusion	134
9.1	Contributions	134
9.2	Future work	135
9.2.1	Auxiliary function for MBR linear regression	135
9.2.2	Error metrics and approximations	136
9.2.3	Sub-word MBR criteria	136
9.2.4	Confidence-driven MBRLR	136
9.2.5	Future discriminative criteria	137
A	MBR criterion optimisation	138
A.1	Preliminary theorems	138
A.2	Definition of MBR auxiliary function	139
A.2.1	Manipulation of MBR auxiliary function	140
A.3	Proof of validity of auxiliary function	150
A.3.1	Preliminary results	151
A.3.2	Proof of Theorem 3	153
A.3.3	Proof of Theorem 4	155
A.3.4	Proof of Theorem 5	155
A.4	Extended Baum-Welch update formulae	158
A.4.1	Means of Gaussian state output distributions	159
A.5	Lower bound on learning rate D	164
B	Experimental Systems	165
B.1	Baseline system	165
B.1.1	Acoustic features	165
B.1.2	Acoustic models	165
B.1.3	Dictionary and language models	166
B.1.4	System operation	166
B.1.5	Training and evaluation datasets	166
B.2	MBR-estimated system operation	167
B.3	MBR parameter estimation details	167

List of Tables

6.1	Analysis of error approximations for substitution, deletion and insertion errors.	70
6.2	Correlation of error approximations with Levenshtein error.	72
6.3	Performance analysis of models yielded by unsmoothed MBR estimation (<i>rt05seval</i> dataset).	76
6.4	Performance analysis of models yielded by I-smoothed MBR estimation (<i>rt05seval</i> dataset).	76
7.1	Relative contribution of temporally overlapping homophones and heteronyms of reference words to the word and phoneme-level MBR criteria.	83
7.2	Contribution of word-end silence errors to the phoneme-level MBR criterion.	85
7.3	Correlation between word, phoneme-sensitive word, silence-sensitive word, phoneme, model and state-level approximated errors.	90
7.4	MBR criterion value and weighting factor for different MBR criteria.	93
7.5	Performance of I-smoothed ($\tau^I = 50$) MBR-estimated models (<i>rt05seval</i> , <i>rt06seval</i> and <i>rt07seval</i> datasets).	93
8.1	Differences between classifiers induced by the word and phoneme-level max- imal frame posterior confidence measures (<i>rt06seval</i> and <i>rt07seval</i> datasets).	117
8.2	Differences between classifiers induced by the word and model-level maximal frame posterior confidence measures (<i>rt06seval</i> and <i>rt07seval</i> datasets).	117
8.3	Differences between classifiers induced by the state and model-level maximal frame posterior confidence measures (<i>rt06seval</i> and <i>rt07seval</i> datasets).	117
8.4	Performance of confidence-weighted MLLR-adapted models (<i>rt06seval</i> and <i>rt07seval</i> datasets).	120

8.5	Performance of ideal confidence-thresholded MLLR (<i>rt07seval</i> and <i>rt07seval</i> datasets).	120
8.6	Differences between ideal confidence measures at the phoneme and model levels (<i>rt06seval</i> and <i>rt07seval</i> datasets).	121
8.7	Differences between ideal confidence measures at the state and model levels (<i>rt06seval</i> and <i>rt07seval</i> datasets).	121
8.8	Differences between ideal confidence measures at the word and model levels (<i>rt06seval</i> and <i>rt07seval</i> datasets).	122
8.9	Performance of standard MLLR and MBRLR (<i>rt06seval</i> and <i>rt07seval</i> datasets).123	
8.10	Differences between classifiers induced by the state and phoneme-level maximal frame posterior confidence measures at thresholds of 0.8 and 0.95 respectively (<i>rt06seval</i> and <i>rt07seval</i> datasets).	127
8.11	Performance of classifiers induced by the state and phoneme-level maximal frame posterior confidence measures at thresholds of 0.8 and 0.95 respectively (<i>rt06seval</i> and <i>rt07seval</i> datasets).	127
8.12	Performance of confidence-weighted MBRLR (<i>rt06seval</i> and <i>rt07seval</i> datasets).127	
8.13	Differences between ideal confidence measures at the phoneme and state levels (<i>rt06seval</i> and <i>rt07seval</i> datasets).	129
8.14	Performance of I-smoothed phoneme-level confidence-thresholded MBRLR (<i>rt06seval</i> and <i>rt07seval</i> datasets).	132
B.1	NIST conference meeting speech evaluation datasets.	167

List of Figures

1.1	Thesis structure.	7
2.1	HMM-based acoustic modelling.	14
2.2	Modelling word sequences with HMMs.	15
2.3	Acoustic model parameter tying.	16
2.4	Overview of ASR classification procedure.	18
3.1	Impact of modelling assumptions on classification.	24
3.2	Maximum margin classifier.	29
4.1	Regression class tree.	37
6.1	Lattice segmentation.	61
6.2	Alignment-based error approximation.	62
6.3	Asymmetry of alignment-based error approximation.	64
6.4	Approximate error in case of insertion.	65
6.5	Approximate error in case of deletion.	66
6.6	Frame error metric.	67
6.7	Frame error metric fails to capture insertion error.	67
6.8	Frame error normalisation in case of deletion error.	68
6.9	Frame error normalisation in case of insertion error.	68
6.10	Minimal symmetrically normalised frame error and approximate minimal symmetrically normalised frame error.	69
6.11	Silence handling using the approximate minimal symmetrically normalised frame error.	71

6.12	Correlation of error approximations with Levenshtein error.	73
6.13	Performance of unsmoothed MBR-estimated models using different error approximations (<i>rt05seval</i> dataset).	75
6.14	Performance of I-smoothed MBR-estimated models using different error approximations (<i>rt05seval</i> dataset).	77
7.1	Comparison of phoneme and word-level MBR criteria in the case of overlapping homophones.	81
7.2	Comparison of phoneme and word-level MBR criteria in the case of overlapping heteronyms.	82
7.3	Additional word-end silence discrimination using the phoneme-level MBR criterion.	84
7.4	Comparison of phoneme and model-level MBR criteria.	88
7.5	Comparison of Levenshtein errors in word, phoneme, model and state-level hypothesis spaces.	89
7.6	Performance of unsmoothed MBR-estimated models using different MBR formulations (<i>rt05seval</i> , <i>rt06seval</i> and <i>rt07seval</i> datasets).	92
7.7	Performance of I-smoothed MBR-estimated models using different MBR formulations (<i>rt05seval</i> , <i>rt06seval</i> and <i>rt07seval</i> datasets).	94
8.1	Calculation of posterior-based confidence measures from a lattice of word alignments.	99
8.2	Calculation of sub-word confidence measures from a model, phoneme and state-aligned lattice.	101
8.3	Standard and confidence-thresholded error approximations.	106
8.4	Evaluation system for adaptation experiments.	111
8.5	Performance of classifiers corresponding to word, phoneme, model and state-level maximal frame posterior confidence measures (<i>rt07seval</i> dataset).	113
8.6	Miss and false alarm rates of classifiers corresponding to word, phoneme, model and state-level maximal frame posterior confidence measures (<i>rt07seval</i> dataset).	114

8.7	Data retained as a function of confidence threshold (<i>rt06seval</i> and <i>rt07seval</i> datasets).	115
8.8	Maximal frame posterior confidence measures for a sample of speech.	116
8.9	Performance of confidence-thresholded MLLR-adapted models as a function of confidence threshold (<i>rt06seval</i> and <i>rt07seval</i> datasets).	119
8.10	Word-level confidence-thresholded MBRLR performance as a function of confidence threshold (<i>rt06seval</i> and <i>rt07seval</i> datasets).	124
8.11	Sub-word level confidence-thresholded MBRLR performance as a function of confidence threshold (<i>rt06seval</i> and <i>rt07seval</i> datasets).	126
8.12	Ideal confidence-thresholded MBRLR performance (<i>rt06seval</i> and <i>rt07seval</i> datasets).	128
8.13	Performance of ideal phoneme-level confidence-thresholded MBRLR with I-smoothing (<i>rt06seval</i> and <i>rt07seval</i> datasets).	131
B.1	2005 AMI meeting speech transcription system.	169
B.2	Details of MBR parameter re-estimation.	170

Abstract

Modern automatic speech recognition (ASR) systems use statistical models of spoken language. These models are typically learned from corpora comprising many hours of transcribed speech. While a variety of machine learning approaches have been applied to this learning task, the optimal learning strategy is unknown. This thesis focusses upon a relatively recent and successful approach, the application of the principle of minimum Bayes risk (MBR) to the estimation and adaptation of acoustic models used in ASR. The aim of the research is to address issues pertaining to the theory, implementation, understanding and performance of MBR acoustic model estimation and adaptation.

The first confronted issue is related to the optimisation of the MBR criterion function in the context of continuous density hidden Markov models (HMMs). Iterative update formulae known as the extended Baum-Welch (EBW) equations are generally used to estimate the parameters of the state output distributions of such HMMs such that the MBR criterion is minimised. Previous justifications of the EBW equations have failed to both guarantee a decrease in the MBR criterion with each iteration and to specify a value of the learning rate constant used in these equations. In this thesis, an auxiliary function for the MBR criterion is presented. Via this auxiliary function, the EBW update equations are derived, and a minimum value for the learning rate constant of these equations is calculated.

The second issue addressed by the thesis concerns the approximation of errors within the implementation of MBR acoustic parameter estimation. Limitations of previously introduced error approximation methods are explained. An alternative error approximation technique which addresses these limitations is presented. Incorporation of this novel error approximation technique yields acoustic models which display significant classification performance improvements over models estimated via previously introduced error approximation methods.

The third issue pertains to the formulation of the MBR criterion, which may be defined using words, phonemes or other sub-word units. The phoneme-level MBR formulation is known as the minimum phone error (MPE) criterion. Previous research has observed small improvements in classification performance when using MPE-estimated acoustic models in place of word-level MBR-estimated acoustic models. This effect is poorly understood. Theoretical arguments and experimental evidence are presented which lend insight into this phenomenon. Additionally, alternative sub-word MBR formulations are proposed, motivated and experimentally evaluated.

The fourth and last issue addressed by this thesis is the performance of acoustic models adapted using unsupervised MBR-based linear regression (MBRLR) adaptation. The theory and implementation of MBRLR acoustic model adaptation is extended by incorporating confidence information into the MBR criterion. This refinement is shown to yield significant classification performance improvements when compared experimentally with standard unsupervised MBRLR adaptation.

Acknowledgements

It is a pleasure to attribute credit to the many people who have contributed directly or indirectly to this work.

I would firstly like to thank Dr Thomas Hain for his supervision throughout my PhD studies.

Secondly, I express gratitude to those who have developed and those who maintain the HTK software toolkit. Without this crucial resource, this thesis would not have been possible.

The members of the Speech and Hearing group at Sheffield University have been, and continue to be, a source of knowledge, friendship and humour. Credit must go to all for making this group both stimulating and nurturing. Sue Harding deserves a particular mention in this regard. Thank you Sue for the coffee breaks, the walks in the Peaks, the ski holidays, the trips to Wales and Scotland, and countless other activities organised. Also, thanks to Sarah Creer for her dedication to the social and intellectual life of the group. Thank you Sarah for the discussion groups, the Friday quizzes, the Christmas party organisation, and for ensuring there was always a cake and a card each birthday.

Penultimately, many thanks to the friends who have provided kindness, humour and distraction from this work, in particular Francois Mairesse.

Lastly, I am hugely indebted to Sarah Creer. Not only for the many hours of dedicated and meticulous proof-reading of this thesis and excellent advice on its presentation. But also simply for her generous and caring nature.

Acronyms

ASR	Automatic speech recognition
CMAP	Conditional maximum a-posteriori
CML	Conditional maximum likelihood
CMN	Cepstral mean normalisation
CVN	Cepstral variance normalisation
EBW	Extended Baum-Welch
EM	Expectation maximisation
HMM	Hidden Markov model
LM	Language model
MAP	Maximum a-posteriori
MBR	Minimum Bayes risk
MBRLR	Minimum Bayes risk linear regression
MCE	Minimum classification error
ML	Maximum likelihood
MLLR	Maximum likelihood linear regression
MPE	Minimum phone error
MPSSWE	Matched pairs sentence segment word error
SAT	Speaker adaptive training
SD	Speaker dependent
SHLDA	Smoothed heteroscedastic linear discriminant analysis
SI	Speaker independent
SNFE	Symmetrically normalised frame error
VTLN	Vocal tract length normalisation
WER	Word error rate

Notation

The denotation of frequently used notation is provided in the following tables.

Matrices, vectors and mathematical notation

\mathbf{x}	General vector
\mathbf{X}	General matrix
\mathbf{X}^\top	Transpose of matrix \mathbf{X}
\mathbf{X}^{-1}	Inverse of matrix \mathbf{X}
$\max_{x \in \mathcal{X}} f(x)$	Maximal value of real function $f(x)$ over the domain \mathcal{X}
$\arg \max_x f(x)$	Value(s) of x for which the real function $f(x)$ is maximal
$\log x$	Natural logarithm of real $x > 0$

HMM and language model parameters

Θ	Set of permitted model parameters
θ	Model parameters (acoustic and/or language model)
$\boldsymbol{\mu}_s$	Mean of output distribution of HMM state s
\mathbf{C}_s	Covariance of output distribution of HMM state s
$\mu_s^{(i)}$	i -th dimension of the mean of of output distribution of HMM state s
$\sigma_s^{2(i)}$	Variance of the i -th dimension of output distribution of HMM state s
$\boldsymbol{\xi}_s$	Extended mean vector of output distribution of HMM state s , $[1, \boldsymbol{\mu}_s^\top]^\top$
\mathbf{W}	Mean transformation matrix

Sequences and sets

\mathbf{o}_1^T	Sequence of T acoustic feature vectors $\mathbf{o}_1\mathbf{o}_2\dots\mathbf{o}_T$
s_1^T	Sequence of T HMM states $s_1s_2\dots s_T$
w_1^N	Sequence of N words, phonemes, models or states $w_1w_2\dots w_N$
\mathcal{S}	Set of state sequences
\mathcal{W}	Set of sequences of words, phonemes, models or states
t	Sequence or frame index
r	Training set utterance index
$\mathbf{o}_t(r)$	t -th member of r -th training set observation sequence $\mathbf{o}_1^{T(r)}$
$\hat{w}_1^{M(r)}$	Correct (or reference) transcription of r -th training set utterance
$\gamma_s(t \hat{w}_1^{M(r)}, \mathbf{o}_1^{T(r)}, \theta)$	Probability $p(s_t = s \mathbf{o}_1^{T(r)}, \hat{w}_1^{M(r)}, \theta)$, the occupancy of state s at time t

Chapter 1

Introduction

The speed and efficiency of spoken language communication make it the preferred medium for many human to human interactions. This predilection has underpinned the introduction of spoken language interfaces between humans and machines in a variety of domains. For example, military fighter aircraft have deployed speech interfaces to reduce the workload on pilots' hands (Weinstein (1991)). Using this application, a pilot is able to issue spoken commands to, for example, adjust radio frequencies and to control cockpit display systems. Another example of a human-machine spoken language interface is the use of speech to control a personal computer, or to dictate electronic documents. Such an interface is available with, for example, Microsoft Vista operating system, and is crucial for users with limited or no use of their hands.

The successful usage of a spoken language interface to a machine depends upon the ability of the machine to understand our speech. This is the task of spoken language understanding, defined as the extraction of meaning from speech. While extremely complex in general, given a constrained task, for example the understanding of a finite set of commands relevant to the cockpit of an aircraft, spoken language understanding by a machine is possible.

Systems with a spoken language interface typically deploy multiple components, including an automatic speech recognition (ASR) component and a semantic interpreter. The task of the ASR component is to translate speech into text, while the semantic interpreter is used to extract meaning from this text. It has been found that successful usage of such interfaces is highly dependent upon the performance of the ASR component of the system (Williams and Young (2007)). Potentially useful spoken language interfaces remain unused due to the unacceptably poor performance of the ASR component. This observation provides ample motivation for the ASR research community.

1.1 Automatic speech recognition

Automatic speech recognition is defined as the translation of a speech waveform into text. If the speech utterance comprises a single word, the task is called isolated word recognition. Recognition of a sequence of words is known as continuous speech recognition. Speech recognition tasks are additionally categorised according to the size of the vocabulary interpreted by the recogniser. For example, in the English language, the task of translating the letters of the Roman alphabet has a vocabulary size of 26. Tasks with a vocabulary size of less than approximately 500 words are classified as small vocabulary, while tasks with a vocabulary size of over 10000 words are large vocabulary. A vocabulary size between 500 and 10000 corresponds with a medium vocabulary task.

Major progress in the field of ASR has been witnessed in the last forty years (Pallett (2003)). Much of this progress has been based on a combination of the following contributions:

- The application of signal processing techniques to define a compact representation of the speech waveform (Itakura and Saito (1970)).
- The introduction of statistical models of spoken language (Jelinek (1976)), and the use of statistical pattern recognition principles to classify the acoustic signal as a word sequence.
- The application of machine learning principles to estimate (Baum et al. (1970)) and adapt (Woodland (2001)) such statistical models.
- The availability of corpora consisting of many hours of transcribed speech data.
- The increased speed of computers, enabling complex learning and classification algorithms to be executed in shorter time intervals.

Despite this progress, the performance of ASR systems compares poorly with human speech recognition for many tasks (Lippmann (1997)). This performance gap has provided supplementary motivation for ASR to remain an active and diverse field of scientific research.

As will be explained in more detail later, the statistical models of spoken language used in modern ASR systems can be separated into an acoustic model and a language model. The acoustic model provides a probabilistic mapping from acoustic information to words contained within the vocabulary of the recogniser. The language model provides prior probabilities associated with (sequences of) in-vocabulary words. This thesis concentrates on the problem of estimation and adaptation of the acoustic model used in modern large vocabulary continuous ASR systems. While a variety of methods have been applied to these tasks, the optimal strategy remains unknown. A successful and relatively recent approach is the application of the principle of minimum Bayes risk (MBR) to the estimation and adaptation of the acoustic model. This thesis focusses upon this method and revises and extends recent research in this area. To help further explain the objectives of the thesis, a brief introduction to statistical pattern classification, acoustic model estimation and acoustic model adaptation follows.

1.2 Pattern classification

The aim of pattern classification (or pattern recognition) is to classify data (or patterns) into one of several predefined categories. For example, a credit card issuer attempts to classify a potential customer as creditworthy or otherwise. In general, a pattern classifier defines a function $f : \mathcal{X} \rightarrow \mathcal{C}$ where \mathcal{X} represents the set of patterns to be classified and \mathcal{C} is a set of categories.

An initial stage of pattern classification is feature extraction. Feature extraction computes a list of properties associated with a pattern. This list is called a feature vector. Only these features are used in subsequent classification. For example, the age and bank balance of a customer may be the facts used in the creditworthiness classification task. Other facts such as hair colour and height may be disregarded. So feature extraction is a mapping $g : \mathcal{X} \rightarrow \mathcal{F}$ where \mathcal{X} is the set of patterns and \mathcal{F} is the feature space. Given a fixed feature extraction method, the pattern classifier is fully defined upon specification of a function $h : \mathcal{F} \rightarrow \mathcal{C}$. Clearly, the feature extraction task is intimately linked with the overall classification task. Pertinent features for one classification task may be irrelevant for another.

Typically some prior constraints are placed upon the function h . Such constraints often limit the functional form or the complexity of the classifier to correspond with the designer's view of the form or complexity of the classification task. For example, the designer of the previously described creditworthiness classifier may specify that a customer is classified as creditworthy if his age is above a threshold or if his bank balance is above a second threshold.

1.2.1 Statistical pattern classification and ASR

Pattern classification can adopt many different paradigms, for example the use of decision trees which map patterns to categories, or the use of nearest neighbour algorithms which infer the category of unseen patterns using a memorised set of correctly classified examples. Statistical pattern classification is another approach to the classification task.

Statistical pattern classification assumes that the generation of patterns and associated categories is governed by a probability distribution. By constructing a statistical model of this distribution, the classifier characterises the posterior probability distribution of categories, given the feature vector associated with a pattern. When a pattern is processed by a statistical pattern classifier, its associated features are compared with the models of each category. One of these categories is then selected and output by the system. The chosen category depends upon:

- the features extracted from the pattern,
- the posterior probability of each category with respect to the statistical models, and
- the decision rule deployed by the system, i.e. the rule used to decide to which category a pattern belongs.

Thus the feature extraction technique, the statistical models and the decision rule define the pattern classification function of a statistical pattern classifier.

The ASR task is an example of pattern classification. The speech waveform corresponding to an utterance is a pattern and the sequence of words associated with the utterance is its category. Modern ASR systems deploy statistical pattern classification using probabilistic models of spoken language. The ASR pattern classification function is therefore dependent upon these statistical models. Indeed, given a fixed feature extraction method and decision rule, the ASR pattern classification function is wholly governed by the statistical models. Estimation and adaptation of these models therefore directly impacts the performance of the resulting classifier. A brief introduction to acoustic model estimation and adaptation is now provided.

1.3 Acoustic model estimation

The acoustic model of an ASR system is typically learned (or estimated) using a set of transcribed speech utterances known as the training data. Learning techniques deployed to date can generally be split into two fundamentally different approaches: generative and discriminative learning. Generative learning seeks to accurately estimate a joint probability distribution over features (associated with patterns) and categories, given the constraints of the chosen statistical model. The aim of discriminative learning contrasts with that of generative learning. Discriminative acoustic model estimation seeks to estimate a classifier which accurately classifies the training data, again given the constraints of the model.

Although generative learning methods have been successful for acoustic model estimation in ASR, significant classification performance improvements have been consistently reported when using discriminatively-learned acoustic models (Juang and Katagiri (1992), Woodland and Povey (2002)). The minimum Bayes risk (MBR) method is an example of a discriminative learning approach. Acoustic models estimated using the MBR technique have not only displayed significant classification performance improvements over generatively-estimated models, but have also yielded significant improvements over models learned using other discriminative approaches (Povey (2003)).

The idea of MBR acoustic model estimation is to alter the acoustic model parameters to minimise a differentiable, real-valued function (often called a criterion) which reflects the amount of classification errors committed on the training dataset. In the context of speech recognition, a classification error may be defined as an erroneous transcription of an utterance. More commonly, a metric known as the word error rate (WER), introduced later, is used to quantify the number of errors associated with a transcription. The WER metric was deployed in the initial formulations of MBR, but a later formulation (Povey (2003)) using a phoneme error rate metric has yielded acoustic models which display superior classification performance.

1.4 Acoustic model adaptation

Speaker dependent (SD) ASR systems are designed to recognise the speech of a single speaker, the speaker used to provide training data for the system. Speaker independent (SI) systems are designed to recognise any speaker. Given an identical amount of training data, an SD system typically displays significantly better performance than an SI system (Woodland (2001)). This is due to a greater mismatch between the test data and the speech models in the case of the SI system.

SI recognition systems use techniques known collectively as speaker adaptation to reduce this mismatch as the system encounters new speakers. A relatively small (in comparison to the volume of data used to train the SI system) quantity of speaker data, known as the adaptation data, is used to adapt the SI system. When the correct transcription of the adaptation data is available the task is referred to as supervised adaptation. An unsupervised adaptation scenario exists if the correct transcription of the adaptation data is unavailable.

One successful approach to speaker adaptation is to adjust the acoustic model, known as acoustic model adaptation. The principles of generative learning have been successfully applied to a range of acoustic model adaptation paradigms (Gauvain and Lee (1994), Leggetter and Woodland (1995), Kuhn et al. (2000)). Discriminative learning methods have also been applied to supervised acoustic model adaptation and some small classification performance improvements over acoustic models adapted using generative learning techniques have been reported (Povey et al. (2003b), Gunawardana and Byrne (2001)). However, relatively little research has focussed on the application of discriminative learning to the unsupervised acoustic model adaptation task (Gunawardana and Byrne (2001), Wang and Woodland (2004)).

1.5 Objectives

This thesis concentrates upon the application of the discriminative MBR technique to the tasks of acoustic model estimation and adaptation in the context of large vocabulary continuous ASR. The primary objectives are to address outstanding issues which exist with current theory, understanding, implementation, and performance of MBR techniques. In this section the issues of concern are summarised and the related thesis objectives explained.

Objective 1: MBR criterion optimisation theory

As will be discussed in due course, the MBR criterion may be optimised using a set of parameter re-estimation formulae known as the extended Baum-Welch equations. These equations are practically useful and there have been several justifications of their use. However, these justifications are unsatisfactory, either because they fail to guarantee the optimisation of the MBR criterion, or because they fail to specify the learning rate used in the parameter update formulae. To address this issue, the first objective of the thesis is to investigate if

an auxiliary function can be specified which both justifies the MBR extended Baum-Welch parameter update equations and specifies the learning rate used in these equations.

Objective 2: MBR error approximation

The implementation of MBR acoustic model estimation involves approximation of errors associated with a set of transcriptions of the training data, as will be discussed later in the thesis. It transpires that a previously introduced approximation technique has some associated limitations. The second objective is to investigate if an error approximation technique can be found which addresses the limitations of the previously proposed method. Additionally, if such a technique can be found, a secondary objective is to evaluate and understand the impact of the novel approximation method upon the performance of the resulting MBR-estimated acoustic models.

Objective 3: Sub-word MBR criteria

The superior performance of phoneme-level MBR-estimated acoustic models over word-level MBR-estimated models is poorly understood. The third thesis objective is to better understand this phenomenon. Some additional objectives arise from questions related to the issue of understanding the superiority of the phoneme-level MBR formulation. One supplementary objective is to investigate if use of other sub-word units is motivated when formulating the MBR criterion. If so, a further objective is to investigate if these other sub-word MBR-estimated acoustic models deliver improved classification performance over word-level and phoneme-level MBR-estimated acoustic models.

Objective 4: Confidence-driven MBR acoustic model adaptation

Relatively few publications have reported classification performance gains (over the equivalent generative adaptation technique) when using a discriminative acoustic model adaptation technique in an unsupervised scenario. The final objective of the thesis is to discover if the incorporation of confidence information into the unsupervised MBR-based acoustic model adaptation technique can enhance the performance of the resulting adapted models.

1.6 Overview

Figure 1.1 represents the structure of the thesis. The boxes show the chapter numbers and titles. The solid arrows reflect a high degree of dependency between the chapters. For example, Chapter 5 builds on the discussion of both Chapter 3 and Chapter 4. Dashed arrows indicate a lesser degree of dependency. For example, although the techniques used in Chapter 6 are also used in Chapter 7, it is possible to read Chapter 7 before Chapter 6.

Chapters 2 through 4 provide background on large vocabulary continuous ASR and previous research into discriminative acoustic model estimation and adaptation. Since MBR-based techniques are the focus of the thesis, more detail is provided upon them in Chapter 5. Chapters 6, 7 and 8 contain the main contributions of the thesis. An overview of the

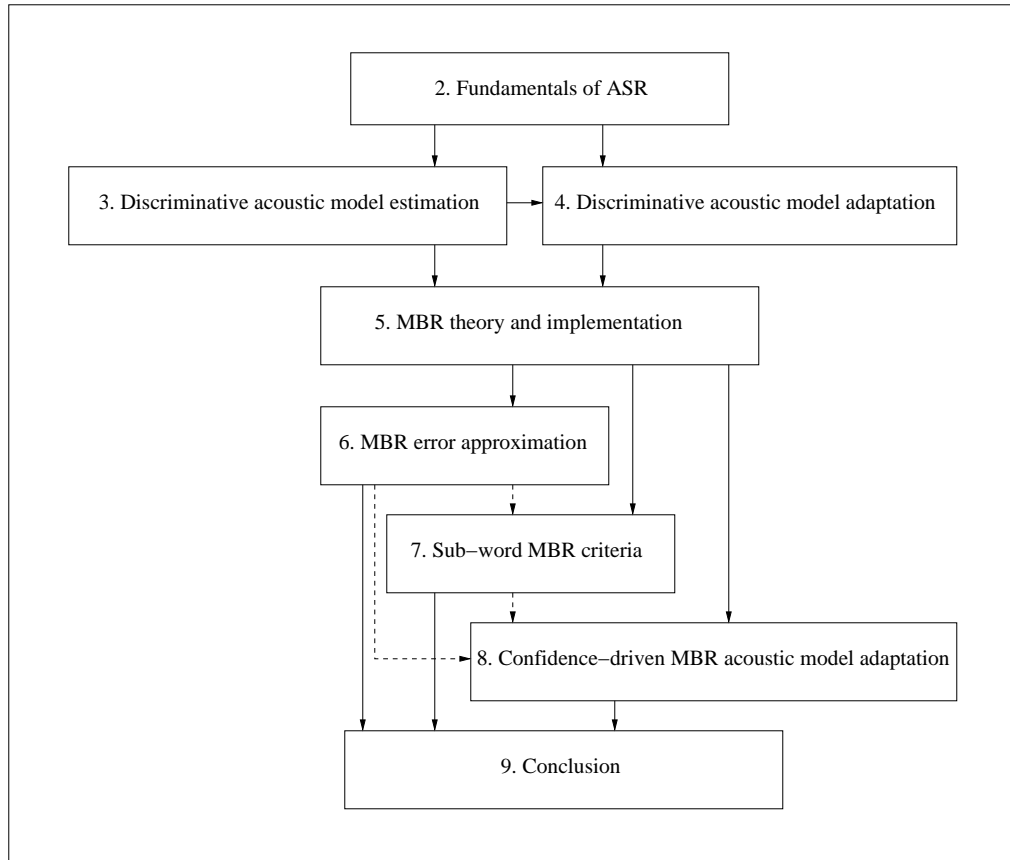


Figure 1.1: *Thesis structure.* Each box represents a chapter. A solid arrow indicates a high degree of dependency between the associated chapters. A dashed arrow indicates a lesser extent of dependency.

content of each chapter follows. Where relevant, this content is linked to the stated thesis objectives.

Chapter 2: Fundamentals of ASR

Chapter 2 provides an introduction to the principles of statistical ASR. The feature extraction, statistical modelling, pattern classification and evaluation procedures used by modern ASR systems are explained. Bayesian decision theory is introduced and the roles of acoustic and language models illustrated. The statistical ASR classification procedure is detailed and the continuous ASR evaluation metric (the word error rate) is defined. A relatively detailed explanation of hidden Markov models (HMM) with regard to acoustic modelling and large vocabulary ASR systems is provided.

Chapter 3: Discriminative acoustic model estimation

Chapter 3 compares generative and discriminative learning approaches and demonstrates how discriminative learning is motivated with regard to acoustic model estimation in the domain of ASR. Previously proposed discriminative acoustic model estimation techniques are reviewed and compared, and the use of the MBR method over alternative discriminative learning approaches is motivated.

Chapter 4: Discriminative acoustic model adaptation

Chapter 4 firstly provides the reader with a structured overview of common speaker adaptation techniques. While these techniques have usually incorporated a generative learning algorithm to adapt the SI system, the success of discriminative acoustic model estimation has prompted research into discriminative acoustic model adaptation. These discriminative adaptation approaches are reviewed in Chapter 4. This chapter provides the background for the work of Chapter 8, which extends current unsupervised MBR-based acoustic model adaptation techniques.

Chapter 5: MBR theory and implementation

Chapter 5 is a detailed explanation of the theory and implementation of MBR acoustic estimation and adaptation. As part of the theoretical discussion, Objective 1 of Section 1.5 is pursued. The techniques and approximations used in parameter optimisation are detailed, as well as methods of improving the generalisation of the resulting acoustic models. This chapter provides a basis for the refinements and extensions described afterwards in the thesis.

Chapter 6: Error approximation

Objective 2, as described in Section 1.5, is the remit of Chapter 6. Previously, Chapter 5 explains how the implementation of MBR acoustic model estimation involves calculation of the error associated with each member of a set of word or phoneme sequences. In the context of large vocabulary continuous speech recognition, this set is relatively large, and exact calculation of these errors becomes prohibitively expensive. Therefore approximations to the word or phoneme error are used in practical implementations of MBR acoustic model estimation and adaptation. Chapter 6 highlights some limitations of previously introduced error approximation methods and introduces an alternative error approximation technique which addresses these limitations.

Chapter 7: Sub-word MBR criteria

Chapter 7 concentrates on Objective 3, and its related objectives, described in Section 1.5. Theoretical arguments and further experimental evidence supporting the use of the phoneme-level MBR formulation are presented. Analysis demonstrates that the superior generalisation of phoneme-level MBR over word-level MBR is attributable partly to differing

treatment of errors related to acoustic models of silence. Additionally, novel acoustic model-level MBR formulations are introduced, motivated and experimentally evaluated.

Chapter 8: Confidence-driven MBR acoustic model adaptation

The work presented in Chapter 8 addresses Objective 4 of Section 1.5. The theory, implementation and evaluation of unsupervised confidence-driven MBR-based discriminative speaker adaptation is presented.

Chapter 9: Conclusion

Chapter 9 summarises the main contributions of this thesis and highlights questions which may be addressed by future research.

Chapter 2

Fundamentals of ASR

The focus of this thesis is the estimation and adaptation of the acoustic model used in ASR systems. It is therefore crucial to understand the role of the acoustic model with regard to the pattern classification task. The purpose of this chapter is to explain the relationships between the feature extraction, statistical modelling and statistical pattern classification methods used by modern ASR systems.

Not only does this chapter provide background on the operation and evaluation of the ASR systems used in the experimental work of this thesis, but also presents information which underpins the arguments presented later in the thesis. The assumptions included in the acoustic model provide motivation for use of discriminative learning methods, as will be explained in Chapter 3. Additionally, the typical parameter tying techniques used within the acoustic model will be used in Chapter 7 to motivate alternative MBR formulations based upon the tied models. An appreciation of the assumptions included in the language model provides further motivation for the use of discriminative learning in Chapter 3. In the same chapter, knowledge of the ASR evaluation method will also be used to argue in favour of use of MBR acoustic model estimation.

The chapter is structured as follows. A brief overview of speech feature extraction techniques is given in Section 2.1. Statistical pattern recognition is then introduced and related to the acoustic and language models used in the ASR task in Section 2.2. Discussion of acoustic and language modelling is found in Sections 2.3 and 2.4 respectively. An overview of the implementation of the ASR classification procedure is provided in Section 2.5, while Section 2.6 defines the metric used to evaluate ASR systems.

2.1 Feature extraction

The feature extraction component of an ASR system maps the speech waveform onto a sequence of feature vectors. This sequence of feature vectors is subsequently used to classify the speech. This initial stage of the overall classification process is also referred to as front end processing.

To apply digital signal processing techniques to the speech waveform, the analogue waveform is firstly converted into a digital signal. This is done via sampling and quantisation of the waveform. Once the digital signal has been obtained, a variety of techniques can be used to extract features which are useful for the speech classification task. These speech analysis techniques usually assume that the characteristics of the speech signal are stationary over a short time period, typically of the order of 25 milliseconds. The resulting features are a representation of the speech signal over this short time period.

Linear predictive coding (LPC) (Itakura and Saito (1970)) and cepstral analysis (Bogert et al. (1963), Oppenheim et al. (1968)) are examples of such speech analysis techniques. The LPC technique is based on the source-filter paradigm of speech production, which models the speech signal as the excitation of a filter, i.e. the vocal tract, with an energy source. The resulting signal is the convolution of the source signal with the filter function. LPC analysis of the speech signal may be interpreted as estimation of the parameters of the source and filter.

An alternative representation called the cepstral representation, or cepstrum, of a digital signal is defined as the inverse discrete Fourier transform of the log magnitude of the spectral representation. The lower-order coefficients of the cepstral feature vector represent the filter. If a periodic excitation signal is present then this will be reflected in the higher-order coefficients of the cepstrum. Thus a reasonably comprehensive range of cepstral coefficients characterise both the source and filter of the speech production system.

Perceptually-motivated versions of the LPC and cepstral representations are often used in speech recognition to emulate the response of the human auditory system to stimuli. Mel-frequency cepstral coefficients (MFCCs) (Davis and Mermelstein (1980)) and perceptual linear prediction coefficients (PLPs) (Hermansky (1990)) are commonly used examples of perceptually-motivated speech representations.

MFCCs are computed in a similar manner to cepstral coefficients but with one additional intermediate step. The frequency representation, i.e. the discrete Fourier transform of the signal, is further filtered through a series of triangular filters distributed in accordance with the non-linear Mel frequency scale (Stevens et al. (1937)). The log magnitude of the output of each of these filters is then used as the input to the inverse discrete Fourier transform to yield the MFCC representation. The Mel frequency scale approximates the mapping between the frequency of a signal and the human auditory percept of pitch.

PLPs are an alternative version of LPC coefficients. The LPC representation can be computed via the power spectrum i.e. the square of the magnitude of the Fourier transform of the signal. The main difference between PLPs and LPCs is that when computing the PLP representation, a supplementary intermediate step is conducted to filter the power spectrum through of a set of filters similar to those used in the MFCC calculation but distributed with respect to the non-linear Bark scale (Zwicker (1961)). Like the Mel scale, the Bark scale approximates the mapping between frequency and the pitch percept.

A set of MFCC or PLP coefficients are therefore used to characterise the speech signal at a particular instant in time. Generally these features are computed at time instants spaced 10 milliseconds apart. So a speech waveform of length 5 seconds will be represented as a sequence of 500 feature vectors. It remains to explain how this sequence of feature vectors

is used to classify a speech waveform as a sequence of words.

2.2 Pattern classification and ASR

Modern speech recognisers perform classification using Bayesian decision theory. This section demonstrates how the use of this statistical pattern classification technique yields a decision rule. In the case of the ASR classification task, the roles of the acoustic and language models, with respect to the implementation of this decision rule, will be explained.

2.2.1 Bayesian decision theory

Bayesian decision theory is a probabilistic approach to classification which, assuming that the posterior probability distribution of classes given feature vectors is known, guarantees the optimal classification performance. Misclassification errors are quantified using a loss function. Suppose that a feature vector \mathbf{x} belongs to class $c_{\mathbf{x}}$ of a set of possible classes \mathcal{C} . Let $\lambda(c_j|c_{\mathbf{x}})$ represent the loss associated with the action of classifying \mathbf{x} as class c_j . One might define this as in Equation 2.2.1, the so-called zero-one loss function.

$$\lambda(c_j|c_{\mathbf{x}}) = \begin{cases} 0 & \text{if } c_j = c_{\mathbf{x}} \\ 1 & \text{otherwise} \end{cases} \quad (2.2.1)$$

Let $h(\mathbf{x})$ be a classification function which maps the feature vector \mathbf{x} onto a class. Given a loss function, the expected misclassification error $R(h)$ of this classification function is the expected value of the loss, given by Equation 2.2.2.

$$R(h) = \int_{\mathbf{x}} \lambda(h(\mathbf{x})|c_{\mathbf{x}})p(\mathbf{x}) d\mathbf{x} \quad (2.2.2)$$

In the above equation, $p(\mathbf{x})$ is the probability distribution over the feature vectors. Equation 2.2.2 may be rephrased by summing over all the classes, as shown in Equation 2.2.3.

$$R(h) = \int_{\mathbf{x}} \sum_{c \in \mathcal{C}} \lambda(h(\mathbf{x})|c_{\mathbf{x}})p(\mathbf{x}, c) d\mathbf{x} \quad (2.2.3)$$

Minimisation of the expected misclassification error (referred to as the overall risk or the expected risk) is usually the aim of classifier design. When using the zero-one loss function, it can be shown that the classifier which minimises the expected misclassification error is the one which selects the class of maximal posterior probability as described by Equation 2.2.4.

$$h(\mathbf{x}) = \arg \max_{c \in \mathcal{C}} p(c|\mathbf{x}) \quad (2.2.4)$$

This decision rule is referred to as the maximum a-posteriori (MAP) decision rule.

2.2.2 Bayesian decision theory and ASR

The zero-one loss function and resulting MAP decision rule are now applied to the classification task of speech recognition. Let \mathbf{o}_1^T represent a sequence of T acoustic feature vectors corresponding to a speech utterance. A speech recogniser classifies \mathbf{o}_1^T as a sequence of words, w_1^N , say.

Application of the MAP decision rule directly is impossible since the class posterior probability $p(w_1^N | \mathbf{o}_1^T)$ is unknown. This probability is therefore estimated using models of spoken language. Using such models the posterior probabilities of many candidate word sequences are estimated. The MAP decision rule is then applied using these estimates of the class posterior probabilities. The MAP decision rule expressed in Equation 2.2.5 where Bayes' theorem has been used to rearrange the right hand side. The symbol θ is used to represent the parameters of the spoken language models.

$$\begin{aligned} h(\mathbf{o}_1^T) &= \arg \max_{w_1^N} p(w_1^N | \mathbf{o}_1^T, \theta) \\ &= \arg \max_{w_1^N} p(\mathbf{o}_1^T | w_1^N, \theta) p(w_1^N | \theta) \end{aligned} \quad (2.2.5)$$

Equation 2.2.5 highlights two aspects of spoken language modelling. The acoustic model provides the likelihood of the model of a specific word sequence given the acoustic data, $p(\mathbf{o}_1^T | w_1^N, \theta)$. The prior probability of each word sequence, $p(w_1^N | \theta)$, is given by the language model. Sections 2.3 and 2.4 describe some standard approaches to acoustic and language modelling respectively.

2.3 Acoustic modelling

Modern ASR systems use hidden Markov models (HMMs) (Jelinek (1976)) to model the sequence of acoustic feature vectors extracted from the speech waveform. An HMM is a model of a stochastic process. This process is represented as a finite set of discrete states which obey the properties of a Markov chain. The difference between an HMM and a Markov chain is that, in the case of an HMM, the state variable is unobservable, hence the use of the term 'hidden' in the model name. However, variables related to the state, called observations, are observable. When using HMMs to model speech, these observations correspond to the feature vectors extracted from the speech waveform.

An HMM comprising 5 states is illustrated in Figure 2.1. Here the states are depicted as circles. Some states (called emitting states) have an associated probability distribution, the state output distribution, which characterises the observed data of the stochastic process. The states labelled s_2 , s_3 and s_4 of Figure 2.1 have an associated output distribution. An HMM has a transition matrix a_{ij} , representing the probability of the stochastic process moving from state i to state j . Transitions are represented by arrows between states in Figure 2.1 and the probability associated with the transition is detailed beside the corresponding arrows. In general, any state transition is possible, including transitions from a

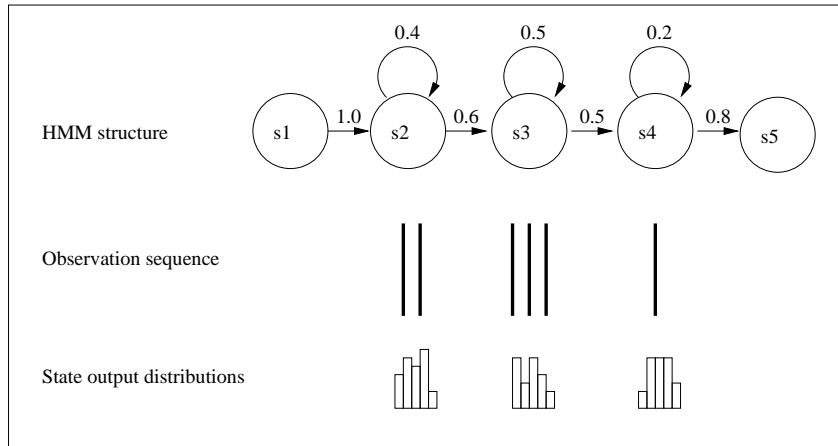


Figure 2.1: *HMM-based acoustic modelling.* The states of the hidden Markov chain are depicted as circles and the arrows indicate the permitted state transitions. The observed features are modelled using output distributions associated with each state.

state to itself. However HMMs used for acoustic modelling often have a left-to-right topology, meaning that transitions from a state s into states further to the left of s within the HMM are assigned zero probability.

If discrete-valued feature vectors are used to represent speech, the state output probability distributions are discrete probability distributions. If continuous-valued feature vectors are used the state output probability distributions are continuous distributions and the HMM is called a continuous density HMM.

An HMM makes assumptions about the stochastic process it models: the first order Markov and conditional independence assumptions. Let the index t represent time. The first order Markov assumption states that the probability of a particular state transition from state s_{t-1} to state s_t , given a state history $s_1 s_2 \dots s_{t-1}$, depends only upon the previous state s_{t-1} , as expressed by Equation 2.3.1.

$$p(s_t | s_1 s_2 \dots s_{t-1}) = p(s_t | s_{t-1}) \quad (2.3.1)$$

The conditional independence assumption states that the probability of an observation \mathbf{o}_t , given a particular state history $s_1 s_2 \dots s_t$ and observation history $\mathbf{o}_1 \mathbf{o}_2 \dots \mathbf{o}_{t-1}$, depends only upon the state at time t and is conditionally independent of both previous states and previous observations. The conditional independence assumption is expressed by Equation 2.3.2.

$$p(\mathbf{o}_t | s_1 s_2 \dots s_t, \mathbf{o}_1 \mathbf{o}_2 \dots \mathbf{o}_{t-1}) = p(\mathbf{o}_t | s_t) \quad (2.3.2)$$

While these assumptions limit the flexibility of an HMM, they permit tractable computations of the probability of a sequence of observations. The model is therefore practical for the purposes of model parameter estimation and the task of pattern classification.

In the domain of speech recognition, distinct HMMs are used to model different speech units. For example, a small vocabulary task such as digit recognition may use distinct HMMs to model the acoustics of each digit in the vocabulary. In the case of large vocabulary speech recognition, HMMs are typically used to model phonemes. A phoneme is the smallest unit in the sound system of a language that serves to distinguish between one word and another. Each word has at least one word to phoneme sequence mapping, specified in a component of the acoustic model called the dictionary. Different word sequences are then modelled by concatenating the models of the comprising phonemes to yield a composite HMM. Figure 2.2 illustrates how the composite HMM representing the word sequence ‘HIGH FIVE’ is constructed from individual phoneme models.

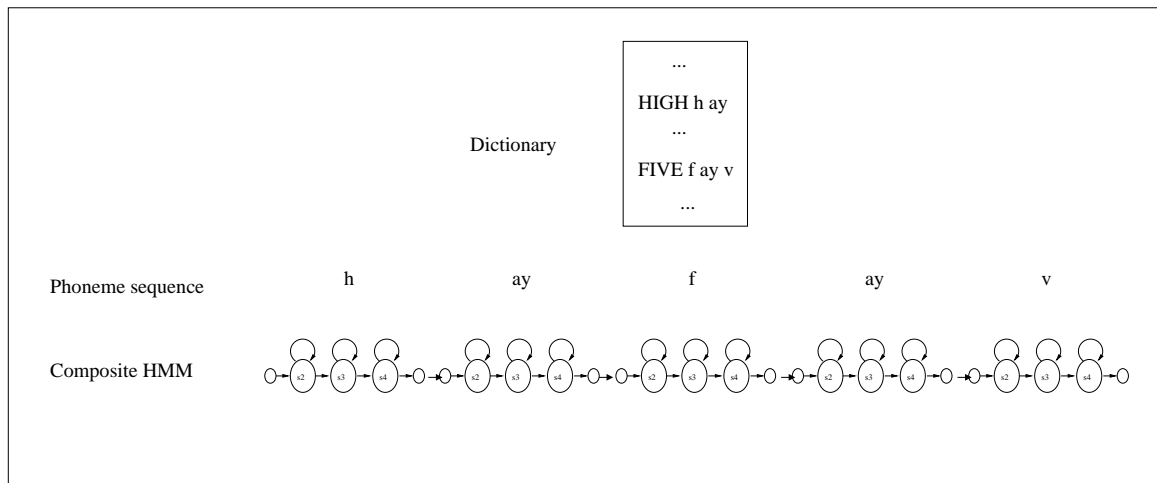


Figure 2.2: *Modelling word sequences with HMMs. The diagram shows the composite HMM for the word sequence ‘HIGH FIVE’. The word sequence is firstly mapped to a phoneme sequence using the dictionary. The phoneme sequence is then modelled by concatenating the HMMs representing each phoneme.*

2.3.1 Coarticulation and acoustic parameter tying

Coarticulation is the effect of adjacent phonemes upon the acoustic realisation of each other. For example, vowels often become nasalised when preceding a nasal consonant. This phonetic variation is typically modelled via context-dependent HMMs. Context-dependent HMMs model the centre phoneme of a phoneme sequence. For example triphone and pentaphone HMMs model the centre phoneme of sequences of three and five phonemes respectively.

Attempting to model every feasible phonetic context results in a large number of model parameters which are difficult to robustly estimate with limited training data. Clustering procedures are therefore used to group the parameters of context-dependent models and

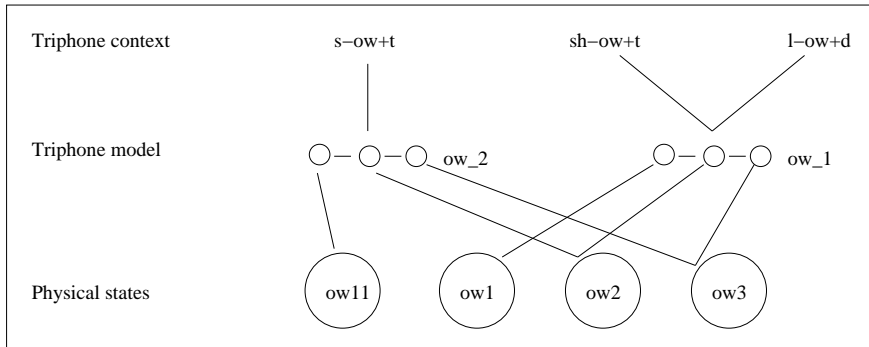


Figure 2.3: *Acoustic model parameter tying. Each triphone context is represented by an HMM. Each HMM is a sequence of states. These states represent a cluster of acoustic contexts, tied using e.g a phonetic decision tree. Those triphone contexts modelled with identical state sequences share the same triphone model.*

consequently reduce the total number of model parameters. The model states are typically clustered via some parameter-tying scheme like a phonetic decision tree (Young et al. (1994)). Consequently, different triphone contexts may be modelled using identical model states, and hence use identical triphone models.

In Figure 2.3, three different triphone contexts for the phoneme ‘ow’ are shown: ‘s-ow+t’, ‘sh-ow+t’ and ‘l-ow+d’. The notation ‘x-y+z’ represents the acoustic context of phoneme ‘y’ when preceded by phoneme ‘x’ and followed by phoneme ‘z’.

As a consequence of state clustering, the triphone contexts ‘sh-ow+t’ and ‘l-ow+d’ share the same 3-state triphone HMM model, labelled ‘ow_1’, while the context ‘s-ow+t’ is modelled by a distinct HMM model ‘ow_2’. Note however that while ‘ow_1’ and ‘ow_2’ are distinct models, their second and third states are identical as a result of state-tying.

2.4 Language modelling

Statistical language models assign a probability $p(w_1^N)$ to a sequence of words w_1^N . This probability may be factorised as shown in Equation 2.4.1.

$$\begin{aligned}
 p(w_1^N) &= p(w_1)p(w_2|w_1)p(w_3|w_1w_2)\dots p(w_N|w_1w_2\dots w_{N-1}) \\
 &= \prod_{k=1}^N p(w_k|w_1^{k-1})
 \end{aligned}
 \tag{2.4.1}$$

This factorisation reduces the language modelling problem to the estimation of the probability $p(w_k|w_1^{k-1})$ of word w_k given a word history w_1^{k-1} . Due to the low frequency of occurrence of longer word sequences, these probabilities are difficult to robustly estimate when the word history becomes very long. One technique which addresses this difficulty is

n -gram language modelling. The assumption used by an n -gram language model is that the probability of the occurrence of a word w_k depends only upon the value of the previous $n - 1$ words. For example a 3-gram (or trigram) language model approximates the probability $p(w_k|w_1^{k-1})$ as $p(w_k|w_{k-1}, w_{k-2})$. An n -gram language model is equivalent to a Markov chain of order $n - 1$, where each word in the vocabulary is represented by a unique state.

The n -gram probabilities $p(w_k|w_{k-n+1}^{k-1})$ are usually estimated using a corpus of text. To gain robust estimates of high order n -gram probabilities (for example 100-grams) one requires a very large corpus. Moreover, while high order n -grams may accurately model the training data they are more likely to overfit this data, and an independent test corpus may be modelled more accurately by lower order n -grams. This is an example of the tradeoff between a model's complexity and ability to generalise.

The issue of robust estimation of higher-order n -grams is typically tackled using techniques such as smoothing and backoff, a thorough treatment of which can be found in Manning and Schütze (1999). The language models used in the experimental work of this thesis include unigrams, bigrams and trigrams i.e. n -grams of order one, two and three respectively. One of the major advantages of deploying n -gram language models for ASR is that the model can easily be integrated into the search algorithm used in classification, as will now be explained.

2.5 ASR search algorithm

Recall that when applying the MAP decision rule to the speech classification task, the word sequence of maximal posterior probability, as described by Equation 2.2.5, is sought. The search procedure performs this task with respect to the language model and acoustic model, as illustrated in Figure 2.4. Given a sequence of acoustic observations \mathbf{o}_1^T , the search algorithm attempts to find the word sequence w_1^N which maximises the joint probability $p(\mathbf{o}_1^T, w_1^N|\theta)$, where the acoustic and language models are represented by θ . Suppose that the set of permissible word sequences is specified in the form of a word grammar (Young et al. (2003)). This grammar can then be expanded into a network of composite HMMs using the dictionary and acoustic models. If no grammar is specified then the search algorithm can dynamically generate a network of composite HMMs using the dictionary, acoustic models and language model. The n -gram language model probabilities can be integrated into this composite HMM as inter-word transition probabilities, expanding the network if necessary to ensure that each network path corresponds to a unique language model context.

Rephrasing the quantity $p(\mathbf{o}_1^T, w_1^N|\theta)$ by summing over the set of all possible state sequences \mathcal{S} of length T , gives Equation 2.5.1. Computation of this sum becomes prohibitively expensive as the length of the observation sequence increases. Since the state sequence which maximises the quantity $p(\mathbf{o}_1^T, w_1^N, s_1^T|\theta)$ tends to dominate this sum, the approximation specified in the second line of Equation 2.5.1 is usually deployed to reduce this

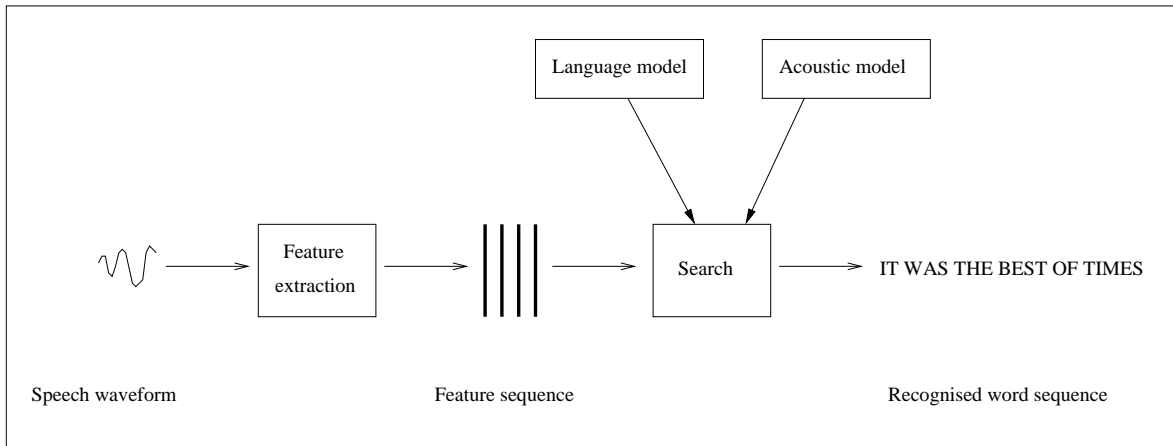


Figure 2.4: *Overview of ASR classification procedure. The feature extraction technique maps the speech waveform to a sequence of feature vectors. The feature vector sequence is subsequently classified as a sequence of words using the search algorithm. The language and acoustic models inform the search procedure.*

computational cost. This is known as the Viterbi approximation.

$$\begin{aligned}
 p(\mathbf{o}_1^T, w_1^N | \theta) &= \sum_{s_1^T \in \mathcal{S}} p(\mathbf{o}_1^T, w_1^N, s_1^T | \theta) \\
 &\approx \max_{s_1^T \in \mathcal{S}} p(\mathbf{o}_1^T, w_1^N, s_1^T | \theta)
 \end{aligned} \tag{2.5.1}$$

The Viterbi algorithm (Viterbi (1967)) is a procedure which guarantees to find the state sequence s_1^T in a composite HMM network which maximises the quantity $p(\mathbf{o}_1^T, w_1^N, s_1^T | \theta)$. Using the Viterbi approximation, the word sequence which corresponds to this state sequence is assumed to be the word sequence which maximises the quantity $p(\mathbf{o}_1^T, w_1^N | \theta)$. This word sequence is consequently the recognised hypothesis.

Even the Viterbi search algorithm proves computationally expensive in the context of large vocabulary continuous ASR. A common technique used to reduce the computation required by the Viterbi algorithm is known as beam pruning or beam search. At each time instant, a beam search disregards (or prunes) state sequences whose likelihood falls below a certain threshold (known as the beam width) of the most likely state sequence at that instant. Small beam widths introduce search errors when the most likely state sequence is discarded at some time instant.

2.6 System evaluation

The most widely used continuous speech recognition evaluation metric is the word error rate (WER). This is based on the Levenshtein distance metric. The Levenshtein distance $L(w_1^N, \hat{w}_1^M)$ is the minimum number of edit operations (insertions, deletions and substitutions) required to transform the recogniser output w_1^N into the correct transcription \hat{w}_1^M . The WER is this Levenshtein distance expressed as a percentage of the number of words in the correct (or reference) transcription.

2.7 Summary

This chapter has provided a concise introduction to the signal processing and pattern recognition techniques used in statistical ASR systems. Some commonly used representations of the speech signal have been presented. The application of Bayesian decision theory to the ASR classification task and the role of acoustic and language models have been explained. A detailed explanation of HMM-based acoustic modelling and an introduction to n -gram language modelling have been provided. An overview of the standard ASR search procedure and a definition of the standard ASR evaluation metric (the word error rate) complete this introduction. Further treatment of these topics can be found in introductory texts on spoken language processing e.g. Huang et al. (2001).

Chapter 3

Discriminative acoustic model estimation

The estimation of the parameters of an acoustic model using a set of transcribed speech utterances is known as acoustic model estimation or training. The optimal training technique is unknown. Consequently, acoustic model estimation is an active research field and a wide variety of techniques have been applied to this task. All of these techniques are based upon one of two fundamentally different approaches to machine learning; generative and discriminative learning.

The purpose of this chapter is to explain the difference between generative and discriminative learning and to motivate the discriminative approach to acoustic model estimation and adaptation. In particular, the discriminative MBR technique will be introduced. Since the main contributions of this thesis (Chapters 6, 7 and 8) are refinements and extensions of the MBR method, use of this particular discriminative technique will be motivated.

The chapter is structured as follows. Section 3.1 explains the principles of generative learning and highlights some theoretical problems with this approach. Section 3.2 motivates the idea of discriminative learning, and contrasts the main discriminative learning approaches which have been used within the domain of ASR. Section 3.3 summarises the content and arguments of the chapter.

3.1 Generative learning

Let \mathcal{Z} denote a set of training examples. Each member \mathbf{z} of \mathcal{Z} is of the form (\mathbf{x}, y) where \mathbf{x} is a feature vector and y is the class to which \mathbf{x} belongs. The generative learning task is defined as the estimation of a joint probability density function $p(\mathbf{z}|\mathcal{Z})$ over the feature vectors and classes, given the training examples \mathcal{Z} . Used in conjunction with Bayesian decision theory, this joint density contains all the necessary information for the task of classification of an unlabelled test example. Thus the estimation of the joint density $p(\mathbf{z}|\mathcal{Z})$ is sufficient for the classification task.

3.1.1 Bayesian inference

Bayesian inference (MacKay (1991)) is a general approach to the task of generative learning of parametric models. Let θ represent the parameters of such a model and let Θ denote the set of all permitted models. Then the distribution $p(\mathbf{z}|\mathcal{Z})$ may be estimated by integrating over the set of all permitted models, as shown in Equation 3.1.1.

$$p(\mathbf{z}|\mathcal{Z}) = \int_{\theta \in \Theta} p(\mathbf{z}|\mathcal{Z}, \theta)p(\theta|\mathcal{Z})d\theta \quad (3.1.1)$$

For many problems, computation of the integral in Equation 3.1.1 is intractable and an approximation known as variational Bayesian inference (Attias, H. (2000)) is often used instead of full Bayesian inference. Due to the complexity of the acoustic models used and the success of simpler approximate techniques, little attention has been paid to Bayesian inference in the domain of ASR (Watanabe et al. (2003), Yu and Gales (2005)). Simpler approximate techniques include maximum a-posteriori and maximum likelihood model estimation.

3.1.2 Maximum a-posteriori model estimation

Maximum a-posteriori (MAP) model estimation can be thought of as an approximation to Bayesian inference where the mode of the posterior distribution of model parameters is used as an approximation to the integral of Equation 3.1.1. The assumption is that the posterior distribution $p(\theta|\mathcal{Z})$ is sharply peaked at its mode and approximates a Dirac delta function. This is expressed mathematically in Equations 3.1.2 and 3.1.3.

$$p(\mathbf{z}|\mathcal{Z}) \approx p(\mathbf{z}|\mathcal{Z}, \theta_{\text{MAP}}) \quad (3.1.2)$$

$$\begin{aligned} \theta_{\text{MAP}} &= \arg \max_{\theta} p(\theta|\mathcal{Z}) \\ &= \arg \max_{\theta} p(\mathcal{Z}|\theta)p(\theta) \end{aligned} \quad (3.1.3)$$

The MAP technique has been successfully employed to estimate HMM acoustic models for ASR (Gauvain and Lee (1994)). Additionally, MAP has proven useful for the task of speaker adaptation, as described later in Section 4.3.1.

3.1.3 Maximum likelihood model estimation

Maximum likelihood (ML) model estimation can be seen as a simplification of the MAP method by using an uninformative (or uniform) prior over the model parameters. The ML parameter estimate is given by Equation 3.1.4.

$$\theta_{\text{ML}} = \arg \max_{\theta} p(\mathcal{Z}|\theta) \quad (3.1.4)$$

In the case of ASR systems, ML schemes estimate the joint distribution $p(\mathbf{o}_1^T, \hat{w}_1^M|\theta)$ of acoustic feature sequences \mathbf{o}_1^T and word sequences \hat{w}_1^M via the parameters θ of the acoustic

and language models. Let $\mathbf{o}_1^{T(r)}$ represent the sequence of feature vectors associated with a training set utterance and let $\hat{w}_1^{M(r)}$ denote the associated transcription. Suppose that there are R such training utterances and that they are statistically independent of each other. Then ML estimation seeks the model parameters θ_{ML} described by Equation 3.1.5. Note that the maximisation of the model likelihood is equivalent to maximisation of the logarithm of the model likelihood. The logarithm is taken for convenience, to convert a product of likelihoods to a sum of log likelihoods.

$$\begin{aligned} \theta_{\text{ML}} &= \arg \max_{\theta} \sum_{r=1}^R \log p(\mathbf{o}_1^{T(r)}, \hat{w}_1^{M(r)} | \theta) \\ &= \arg \max_{\theta} \left[\sum_{r=1}^R \log p(\mathbf{o}_1^{T(r)} | \hat{w}_1^{M(r)}, \theta) + \sum_{r=1}^R \log p(\hat{w}_1^{M(r)} | \theta) \right] \end{aligned} \quad (3.1.5)$$

In the case of acoustic modelling, only the first summation term in Equation 3.1.5 (i.e. the likelihood of the acoustic model) is maximised. The second summation is maximised in ML-based language modelling. Acoustic model estimation involves adjustment of the parameters of the HMMs described in Section 2.3. These parameters are the state transition probabilities and the parameters of the output distribution of each state.

In the case of the acoustic model, the HMMs contain unobserved variables, namely the sequence of states corresponding to a particular sequence of observations. Consequently, the maximisation of the model likelihood is not entirely straightforward. However, an iterative scheme known as the expectation-maximisation (EM) algorithm (Dempster et al. (1977)) can be used to find parameters which locally maximise the likelihood of the acoustic model. With each iteration of this algorithm the model likelihood is guaranteed not to decrease.

The Baum-Welch algorithm (Baum et al. (1970)) is an instance of the EM algorithm used for the estimation of HMM model parameters. The Baum-Welch algorithm uses a procedure called the forward-backward algorithm to compute the probability of a particular state s being present in the hidden state sequence $s_1^{T(r)}$ at a particular frame t , given an observation sequence $\mathbf{o}_1^{T(r)}$ and a composite HMM representing the word sequence $w_1^{N(r)}$ with parameters θ . This probability, $p(s_t = s | \mathbf{o}_1^{T(r)}, \hat{w}_1^{M(r)}, \theta)$, is called the occupancy of state s at time t and represented by the symbol $\gamma_s(t | \hat{w}_1^{M(r)}, \mathbf{o}_1^{T(r)}, \theta)$. These occupancies are then used to formulate the expected value of the model likelihood (the expectation step). The model parameters are subsequently adjusted to maximise this expected value (the maximisation step).

In the case of HMMs with continuous density Gaussian mixture output distributions, Baum-Welch estimates of the transition probabilities, mixture weights, mixture means and covariances are used. For the purposes of illustration, the mean and covariance of a single-mixture Gaussian output distribution are provided here. The estimate of the mean $\hat{\boldsymbol{\mu}}_s$ of a particular HMM state s is given by Equation 3.1.6 where $\mathbf{o}_t(r)$ is the t -th member of the

sequence of observations $\mathbf{o}_1^{T(r)}$.

$$\hat{\boldsymbol{\mu}}_s = \frac{\sum_{r=1}^R \sum_{t=1}^{T(r)} \gamma_s(t|\hat{w}_1^{M(r)}, \mathbf{o}_1^{T(r)}, \theta) \mathbf{o}_t(r)}{\sum_{r=1}^R \sum_{t=1}^{T(r)} \gamma_s(t|\hat{w}_1^{M(r)}, \mathbf{o}_1^{T(r)}, \theta)} \quad (3.1.6)$$

The estimate of the covariance $\hat{\mathbf{C}}_s$ of HMM state s is given by Equation 3.1.7 where $\boldsymbol{\mu}_s$ is the current (non-updated) mean.

$$\hat{\mathbf{C}}_s = \frac{\sum_{r=1}^R \sum_{t=1}^{T(r)} \gamma_s(t|\hat{w}_1^{M(r)}, \mathbf{o}_1^{T(r)}, \theta) (\mathbf{o}_t(r) - \boldsymbol{\mu}_s) (\mathbf{o}_t(r) - \boldsymbol{\mu}_s)^\top}{\sum_{r=1}^R \sum_{t=1}^{T(r)} \gamma_s(t|\hat{w}_1^{M(r)}, \mathbf{o}_1^{T(r)}, \theta)} \quad (3.1.7)$$

3.1.4 Problems with generative learning

Estimation of the joint density of patterns and categories $p(\mathbf{z}|\mathcal{Z})$ is appealing because it leads to informative model estimation. However, it is not clear that this is efficient usage of the training data. Indeed if the goal of the learning process is to construct a classifier then it may be more efficient to directly learn the classification function which maps patterns to classes instead of indirectly learning this function via the intermediate estimation of this joint probability distribution.

Moreover, if the models used are incorrect, i.e. if the family of probability distributions described by the models does not include the true data distribution, then the techniques derived from Bayesian inference such as MAP and ML cannot estimate the correct generative distribution, even with unlimited training data. Consequently the performance of the resulting classifier is compromised. This idea is illustrated in Figure 3.1. Here the modelled posterior probability distributions of two classes are shown, each modelled by a single Gaussian mixture component distribution. The true posterior distribution of class 1 is a single Gaussian, while the true posterior of class 2 is a mixture of two Gaussians. Given unlimited training data, generative training algorithms such as ML are capable of accurately estimating the model of the posterior distribution of class 1. However the true distribution of class 2, depicted by the bimodal dashed curve in Figure 3.1, can only be approximated by the single mixture distribution shown by the dotted curve on the left hand side of the figure. The ideal decision boundary of a classifier is shown, the point where the true posterior distributions are equal. The MAP decision boundary used by the classifier resulting from the generative model is also shown. Clearly this classifier is suboptimal since its decision boundary deviates from the ideal decision boundary.

The assumptions included in HMMs, discussed in Section 2.3, and those included in n -gram language models, discussed in Section 2.4, render these models incapable of capturing the true posterior probability distribution of word sequences in spoken language. Thus generative learning of HMM acoustic models may be suboptimal. This is sufficient motivation to consider alternative learning techniques for the purpose of HMM acoustic model estimation.

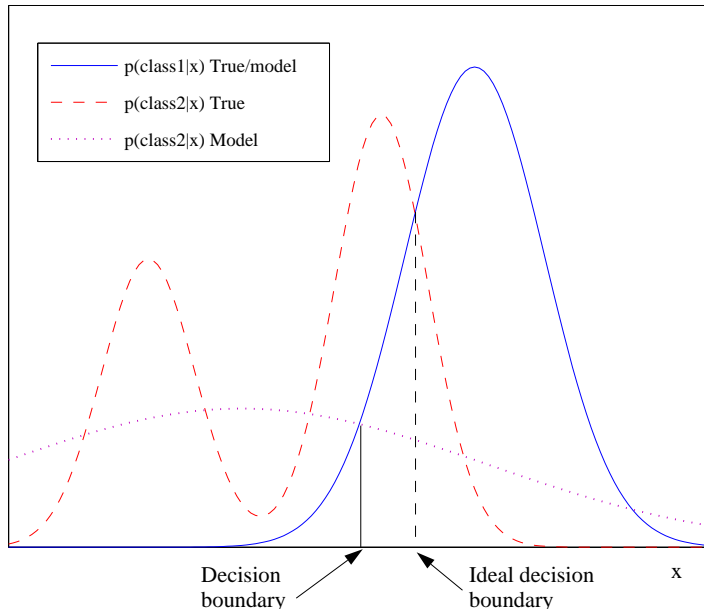


Figure 3.1: *Impact of modelling assumptions on classification. The posterior probability distributions of two classes are each modelled by a single mixture Gaussian distribution. The true posterior distribution of class 1 is a single Gaussian. The true posterior of class 2 is a mixture of two Gaussians, depicted by the bimodal dashed curve. This is approximated with the single mixture distribution shown by the dotted curve on the left hand side. This approximation leads to a decision boundary which differs from the ideal decision boundary.*

3.2 Discriminative learning

Discriminative learning approaches differ from generative learning in that they directly seek model parameters which exhibit good classification performance. The main difference between discriminative and generative learning approaches is that discriminative learning emphasises the accurate estimation of class decision boundaries instead of the accurate estimation of the joint probability distribution of features and classes. In contrast to generative learning, discriminative learning capitalises upon knowledge of the classification task.

Some successful classification models, for example support vector machines (SVMs) (Vapnik (1998)), adopt a classification paradigm which fundamentally differs from Bayesian decision theory. These models directly estimate the class decision boundaries without estimation of class posterior distributions. However, these models lack several desirable properties of an HMM e.g. the capacity to elegantly model sequences of acoustic features of varying length and the ability to naturally represent an unbounded number of word sequences. For

these reasons, relatively little research has been conducted into the use of such discriminative acoustic models for the task of speech recognition (Venkataramani et al. (2007)). The majority of research into discriminative acoustic model estimation has maintained the structure of the HMM and the application of Bayesian decision theory to the classification task, while altering the HMM parameters to satisfy discriminative criteria which fundamentally differ from the generative ML and MAP criteria. These discriminative criteria are introduced in this section.

3.2.1 Conditional Bayesian inference

Conditional learning differs from generative learning in that it directly estimates the distribution used in classification i.e. the posterior distribution $p(y|\mathbf{x}, \mathcal{Z})$ of classes y given the features \mathbf{x} and the training data \mathcal{Z} . A general formalism of conditional learning called conditional Bayesian inference is presented in Jebara (2002). Again, let θ represent the parameters of a model and Θ represent the set of all permitted models. Then the distribution $p(y|\mathbf{x}, \mathcal{Z})$ is estimated by integrating over all these models as shown in Equation 3.2.1.

$$\begin{aligned} p(y|\mathbf{x}, \mathcal{Z}) &= \int_{\theta \in \Theta} p(y|\mathbf{x}, \mathcal{Z}, \theta) p(\theta|\mathbf{x}, \mathcal{Z}) d\theta \\ &= \int_{\theta \in \Theta} p(y|\mathbf{x}, \theta) p(\theta|\mathcal{Z}) d\theta \end{aligned} \quad (3.2.1)$$

Let \mathcal{X} represent the set of feature vectors in the training set and let \mathcal{Y} represent the corresponding set of classes. Then Equation 3.2.1 may be rewritten in terms of the conditional distribution $p(\mathcal{Y}|\mathcal{X}, \theta)$, as shown in Equation 3.2.2.

$$\begin{aligned} p(y|\mathbf{x}, \mathcal{Z}) &= \int_{\theta \in \Theta} p(y|\mathbf{x}, \theta) p(\theta|\mathcal{Z}) d\theta \\ &= \int_{\theta \in \Theta} p(y|\mathbf{x}, \theta) \frac{p(\mathcal{Y}|\mathcal{X}, \theta) p(\mathcal{X}|\theta) p(\theta)}{p(\mathcal{X}, \mathcal{Y})} d\theta \\ &= \int_{\theta \in \Theta} p(y|\mathbf{x}, \theta) \left[\frac{p(\mathcal{Y}|\mathcal{X}, \theta) p(\mathcal{X}) p(\theta)}{p(\mathcal{X}, \mathcal{Y})} \right] d\theta \end{aligned} \quad (3.2.2)$$

Note that $p(\mathcal{X}|\theta)$ is identical to $p(\mathcal{X})$ because, in the case of conditional learning, the model parameters θ are independent of the features \mathbf{x} . The parameters characterise the conditional distribution $p(y|\mathbf{x})$ but do not characterise the distribution $p(\mathbf{x})$.

Conditional MAP and conditional ML

As in the case of Bayesian inference, the integral over all considered models described by Equation 3.2.1 is usually intractable. Approximations analagous to MAP and ML exist and are called conditional MAP (CMAP) and conditional ML (CML) respectively. The CMAP parameter estimate θ_{CMAP} is the mode of the bracketed term inside the integral of Equation 3.2.2, as expressed by Equation 3.2.3.

$$\theta_{\text{CMAP}} = \arg \max_{\theta} p(\mathcal{Y}|\mathcal{X}, \theta) p(\theta) \quad (3.2.3)$$

The CML parameter estimate is the CMAP estimate with uninformative prior as expressed by Equation 3.2.4.

$$\theta_{\text{CML}} = \arg \max_{\theta} p(\mathcal{Y}|\mathcal{X}, \theta) \quad (3.2.4)$$

Both CML and CMAP have been successfully employed in the field of ASR for the purpose of acoustic model estimation (Nadas et al. (1988), Normandin (1991), Woodland and Povey (2002), Povey (2003)) and acoustic model adaptation (Gunawardana and Byrne (2001), Povey et al. (2003b), Povey et al. (2003a)). Note that, in the case of acoustic model estimation, CML is often referred to as the maximum mutual information (MMI) technique, while CMAP corresponds to smoothed versions of MMI.

A particular example of the benefit of conditional Bayesian learning over generative Bayesian learning is provided in Chapter 2 of Jebara (2002). In the case of speech recognition, an example of the improved robustness (to an imperfect language model) of CML over ML acoustic model estimation is provided in Nadas et al. (1988). Since conditional learning is not the focus of this thesis, the exploration of these examples is left to the interested reader.

CML-estimated acoustic models offer classification improvements over ML-estimated models, as evidenced by the results of large vocabulary conversational telephone speech recognition experiments conducted using the Switchboard corpus (Godfrey et al. (1992)) in Woodland and Povey (2002). However the CML criterion does not explicitly attempt to reduce the misclassification rate of the resulting classifier. Discriminative estimation criteria which do address the misclassification error are now introduced. These criteria may be viewed as smoothed instances of a measure known as the empirical risk.

3.2.2 Empirical risk

Section 2.2.1 introduced the loss function $\lambda(c_j|c_{\mathbf{x}})$ as a measure of the classification error when class c_j is assigned to a feature vector whose true class is $c_{\mathbf{x}}$. Given such a loss function and a classification function $h(\mathbf{x})$, where \mathbf{x} is a feature vector, the expected misclassification error of a classifier is given by Equation 2.2.2.

The expected misclassification error may be regarded as the optimal criterion for classifier design. However it is generally not useful since the joint distribution of features and classes, $p(\mathbf{x}, c)$, is usually unknown. The empirical risk $R_{\text{emp}}(h)$ approximates the expected misclassification error using a finite set of labelled training examples, as described by Equation 3.2.5.

$$R_{\text{emp}}(h) = \frac{1}{R} \sum_{r=1}^R \lambda(h(\mathbf{x}_r)|c_r) \quad (3.2.5)$$

In the above equation, c_r is the correct class of feature vector \mathbf{x}_r . Selection of a classification function which minimises the $R_{\text{emp}}(h)$ is known as empirical risk minimisation (ERM). Under certain conditions, (Vapnik (1998)) the empirical risk is shown to converge to the expected risk as the number of training examples tends to infinity. Under more strict conditions, the classification function which minimises the empirical risk is shown to coincide

with the function which minimises the expected misclassification error as the number of training examples tends to infinity.

In the context of ASR, where continuous density HMM acoustic models and Bayesian decision theory are deployed using a discrete-valued loss function (e.g. the zero-one loss or Levenshtein error function), it can be shown that the empirical risk is not a continuous, and hence not a differentiable, function of the model parameters. Therefore the empirical risk criterion is impractical when using, for example, gradient-descent optimisation techniques to estimate the model parameters. Such problems have been addressed by the introduction of differentiable functions which approximate the empirical risk criterion. The minimum classification error and minimum Bayes risk criteria, presented in Sections 3.2.3 and 3.2.5 respectively, are examples of such methods.

3.2.3 Minimum classification error

The minimum classification error (MCE) criterion, formulated in Juang and Katagiri (1992), is a smoothed version of the training set misclassification rate. The formalism uses discriminant functions $g_i(\mathbf{x}|\theta)$ for each class c_i and training example \mathbf{x} . When applied to speech recognition (Reichl and Ruske (1995)), the discriminant functions are as defined in Equation 3.2.6, where θ represents the model parameters.

$$g_i(\mathbf{x}|\theta) = \log p(\mathbf{x}, c_i|\theta) \quad (3.2.6)$$

The misclassification measure $d(\mathbf{x}|\theta)$ is defined in Equation 3.2.7, where c_i is the correct class label and $\bar{g}_i(\mathbf{x}|\theta)$ is defined by Equation 3.2.8.

$$d(\mathbf{x}|\theta) = -g_i(\mathbf{x}|\theta) + \bar{g}_i(\mathbf{x}|\theta) \quad (3.2.7)$$

$$\bar{g}_i(\mathbf{x}|\theta) = \log \left[\left[\sum_{c_j \neq c_i} \exp(\eta \log p(\mathbf{x}, c_j|\theta)) \right]^{1/\eta} \right] \quad (3.2.8)$$

In Equation 3.2.8, η is a positive constant. Small values of η increase the dominance of classes with low discriminant function values within the misclassification measure. As η increases, it can be shown that $\bar{g}_i(\mathbf{x}|\theta)$ tends to $\max_{j \neq i} \log p(\mathbf{x}, c_j|\theta)$, so $\bar{g}_i(\mathbf{x}|\theta)$ is referred to as the softmax function (Bridle (1989)). Note that when the number of classes is large, as in the case of large vocabulary ASR, Equation 3.2.8 must be approximated. Typically an N-best list or a word lattice is used in such an approximation.

The MCE criterion, $R_{\text{MCE}}(\theta)$, is given by Equation 3.2.9 where $l(d(\mathbf{x}_r|\theta))$ is the MCE loss function and, as beforehand, R is the number of training examples.

$$R_{\text{MCE}}(\theta) = \frac{1}{R} \sum_{r=1}^R l(d(\mathbf{x}_r|\theta)) \quad (3.2.9)$$

A typical choice of the function l is the sigmoid, given by Equation 3.2.10.

$$l(d(\mathbf{x}_r|\theta)) = \frac{1}{1 + e^{-\alpha d(\mathbf{x}_r|\theta)}} \quad (3.2.10)$$

The sigmoid function is a smoothed approximation to the zero-one loss function; positive misclassification measures ($d(\mathbf{x}_r|\theta) > 0$) result in a loss function closer to 1 while negative misclassification measures result in a loss function closer to 0. The constant α is a positive real number which controls the gradient of the sigmoid. In the limits of $\alpha \rightarrow \infty$ and $\eta \rightarrow \infty$, the MCE criterion tends to the utterance-level training set misclassification rate. MCE model estimation seeks the parameters which minimise $R_{\text{MCE}}(\theta)$.

A comparison of the MCE and CML criteria, including a comparison of different criterion optimisation techniques, is presented in Schluter et al. (2001). When deployed for the purpose of a small vocabulary continuous speech recognition task using the *SieTill* corpus of German digits (Eisele et al. (1996)), some small classification performance improvements are observed when using MCE-estimated acoustic models in place of CML-estimated acoustic models.

One theoretical issue arises when the MCE criterion is applied to the task of continuous speech recognition. The transcription of an entire utterance, i.e. a sequence of words, is used as the unit of classification in the MCE criterion. However, utterance-level classification is in conflict with the standard continuous speech evaluation metric, the word error rate. When using the WER metric, some utterance misclassifications incur a greater loss than others. The minimum Bayes risk criterion, introduced in Section 3.2.5, addresses this conflict. Before explaining the MBR method, maximum margin estimation is introduced and related to the MCE technique.

3.2.4 Maximum margin estimation

The idea of maximum margin estimation is not simply to find the parameters which exhibit the best classification performance on the training data. As illustrated by the dashed lines in Figure 3.2, there may be many different classifiers which perfectly classify the training examples. Maximum margin estimation not only seeks model parameters which exhibit good classification performance on the training dataset, but also seeks those parameters which maximise the distance between decision boundaries and training examples. The minimum distance between the training data points and the decision boundary is called the margin, and maximum margin estimation aims to maximise this measure. The solid line in Figure 3.2 describes the classifier of maximum margin for the dataset shown.

The computational learning theory which underpins maximum margin estimation shows that the generalisation of a classifier improves as the margin increases (Vapnik (1998)). Therefore, attempts have been made to estimate the parameters of continuous density HMMs with the maximum margin criterion (Li et al. (2005)). The maximum margin criterion uses those training examples which are correctly classified. Of those correctly classified examples, only those which have a closely competing incorrect class are selected. Let \mathcal{T} denote the entire training dataset. Then the subset \mathcal{M} of the training dataset chosen for maximum margin estimation is given by Equation 3.2.11.

$$\mathcal{M} = \{(\mathbf{x}_r, c_r) \in \mathcal{T} : 0 \leq g_{c_r}(\mathbf{x}|\theta) - \max_{c_i \neq c_r} g_i(\mathbf{x}|\theta) \leq \epsilon\} \quad (3.2.11)$$

Here c_r is the correct class associated with features \mathbf{x}_r , $g_i(\mathbf{x}|\theta)$ is the discriminant function of

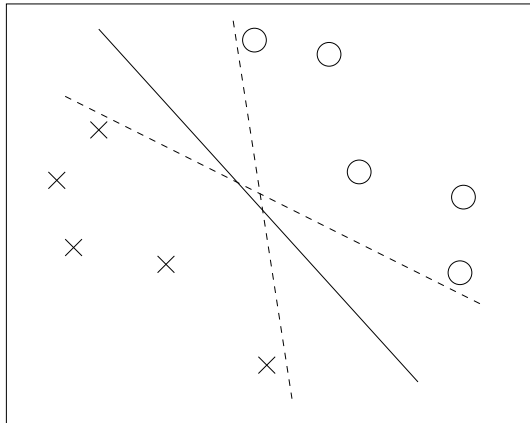


Figure 3.2: *Maximum margin classifier. The crosses and circles represent training examples of different classes. The dashed lines are the decision boundaries of classifiers which perfectly classify this training data. The solid line represents the decision boundary of the maximum margin classifier.*

Equation 3.2.6 and ϵ is a parameter which defines how close the likelihood of the competing class must be to the likelihood of the correct class in order to be included in set \mathcal{M} . Then the margin $M_{\text{marg}}(\theta)$ of this set is the minimum difference between the discriminant function of the correct class and the largest value of the discriminant function of the incorrect classes. This is expressed by Equation 3.2.12.

$$M_{\text{marg}}(\theta) = \min_{(\mathbf{x}_m, c_m) \in \mathcal{M}} \left[g_{c_m}(\mathbf{x}_m | \theta) - \max_{c_i \neq c_m} g_i(\mathbf{x}_m | \theta) \right] \quad (3.2.12)$$

The maximisation operation on the right hand side of Equation 3.2.12 means that $M_{\text{marg}}(\theta)$ is a non-differentiable function of the parameters. In a similar manner to the MCE criterion, the maximisation is therefore replaced with the softmax function to yield the differentiable maximum margin criterion $R_{\text{MME}}(\theta)$ of Equation 3.2.13. Here $\bar{g}_{c_m}(\mathbf{x}_m | \theta)$ is the function defined in Equation 3.2.8.

$$R_{\text{MME}}(\theta) = \min_{(\mathbf{x}_m, c_m) \in \mathcal{M}} [g_{c_m}(\mathbf{x}_m | \theta) - \bar{g}_{c_m}(\mathbf{x}_m | \theta)] \quad (3.2.13)$$

Maximum margin parameter estimation selects those parameters which maximise the criterion $R_{\text{MME}}(\theta)$. In the case of HMMs with Gaussian mixture output distributions, the quantity $g_{c_m}(\mathbf{x}_m | \theta) - \bar{g}_{c_m}(\mathbf{x}_m | \theta)$ is unbounded and attempts have been made to overcome the resulting maximisation problem. For example, normalisation of the criterion by the correct-class discriminant function has been introduced in Lui et al. (2005) to ensure that the criterion is bounded. More recently, to ensure that the criterion has a well-defined maximum, some additional constraints have been placed upon the model parameters (Jiang et al. (2006)).

The maximum margin estimation and the MCE technique are similar, and differ only in the way they select training examples. The MCE method deploys a summation over all training examples and utilises a sigmoid function to de-emphasise the impact of examples far from the decision boundary upon the criterion. The maximum margin method selects only the correctly classified example closest to the decision boundary in the definition of the criterion.

Maximum margin estimation of acoustic models is theoretically appealing and significant gains over the classification performance of MCE-estimated models have been reported (Jiang et al. (2006), Yu et al. (2008)). The first of these publications reports significant classification gains over the performance of MCE-estimated models on both an isolated English alphabet letter (Cole et al. (1990)) and the TIDIGITS connected digit (Leonard (1984)) small vocabulary recognition tasks. The second publication confirms that significant classification gains over the performance of MCE-estimated models are yielded for the TIDIGITS task, and also that maximum-margin-estimated models yield significant classification performance gains over MCE-estimated models for a large vocabulary telephone speech transcription task using a Microsoft-internal speech database.

Despite this encouraging progress, there remain several arguments against the use of the maximum margin criterion. Firstly, it may be argued that the reduction of the training examples to only those in set \mathcal{M} is inefficient usage of the data. Secondly, and similarly to the objection raised to MCE training in Section 3.2.3, the WER evaluation criterion is disregarded. When using MCE or maximum margin estimation, all misclassifications are treated equally for the purpose of model estimation but not for the purpose of model evaluation. The first limitation is addressed somewhat by the large-margin-based estimation criteria introduced in Li et al. (2006) and Yu et al. (2008), where misclassified training examples contribute to modified margin-based criteria.

3.2.5 Minimum Bayes risk

As introduced in Section 2.6, the standard evaluation metric for an ASR system is the word error rate, based upon the Levenshtein distance between the correct word sequence and the recognised hypothesis output by the system. The MBR criterion (also referred to as the overall risk criterion), introduced in Kaiser et al. (2002), incorporates the evaluation metric into its definition, given by Equation 3.2.14.

$$R_{\text{MBR}}(\theta) = \frac{1}{R} \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} p(w_1^N | \mathbf{x}_r, \theta) L(w_1^N, \hat{w}_1^{M(r)}) \quad (3.2.14)$$

The set \mathcal{W} comprises all possible word transcriptions of the training utterance \mathbf{x}_r , $\hat{w}_1^{M(r)}$ is the correct transcription of \mathbf{x}_r and $L(w_1^N, \hat{w}_1^{M(r)})$ is the Levenshtein distance between the correct transcription and hypothesis w_1^N . The set \mathcal{W} is called the hypothesis space. The criterion may be interpreted as the empirical risk of the recogniser described by the classification function $h(\mathbf{x}|\theta)$ using the loss function $\lambda_{\text{MBR}}(h(\mathbf{x}|\theta)|\hat{w}_1^{M(r)})$ given by Equation

3.2.15.

$$\lambda_{\text{MBR}}(h(\mathbf{x}_r|\theta)|\hat{w}_1^{M(r)}) = \sum_{w_1^N \in \mathcal{W}} p(w_1^N|\mathbf{x}_r, \theta)L(w_1^N, \hat{w}_1^{M(r)}) \quad (3.2.15)$$

Minimisation of the MBR criterion, $R_{\text{MBR}}(\theta)$, is motivated by observing that this explicitly attempts to reduce misclassifications of the training data, unlike the CML criterion introduced in Section 3.2.1. Moreover, unlike the MCE and maximum margin criteria, which use an utterance-level misclassification measure, the MBR criterion uses the Levenshtein evaluation metric to quantify misclassifications. Thus there is no conflict between the criteria used to estimate and evaluate models.

There are several publications which compare the classification performance of MBR-estimated acoustic models with other discriminatively-estimated acoustic models. Consistent evidence of the superiority of MBR-estimated acoustic models over CML-estimated acoustic models for large vocabulary recognition tasks (broadcast news and Switchboard telephone conversations) is presented in Povey (2003) and Doumpiotis and Byrne (2004). In Macherey et al. (2005), no significant difference in classification performance is found between MCE and MBR-estimated acoustic models, where the performance of the models is compared for the Wall Street Journal large vocabulary speech recognition task.

Minimum phone error

One of the main objectives of this thesis (Chapter 7) involves the investigation of sub-word formulations of the MBR criterion. The minimum phone error criterion (MPE), introduced in Povey (2003), is an example of one such formulation which has been the focus of much attention in the field of acoustic model estimation. The MPE criterion differs from the word-level MBR criterion introduced above in only one way; it uses a loss function corresponding to phoneme-level misclassification errors instead of word-level misclassification errors. This alternative loss function $\lambda_{\text{MPE}}(h(\mathbf{x}|\theta)|\hat{w}_1^M)$ is given by Equation 3.2.16, where \hat{w}_1^M is the correct phoneme-level transcription¹ of the acoustic features \mathbf{x} , \mathcal{P} is the set of all phoneme-level transcriptions of \mathbf{x} and $L(w_1^N, \hat{w}_1^M)$ is the Levenshtein distance between the correct transcription and the hypothesis w_1^N .

$$\lambda_{\text{MPE}}(h(\mathbf{x}|\theta)|\hat{w}_1^M) = \sum_{w_1^N \in \mathcal{P}} p(w_1^N|\mathbf{x}, \theta)L(w_1^N, \hat{w}_1^M) \quad (3.2.16)$$

Experimental results in Povey (2003) have reported that the MPE criterion yields acoustic models which display superior classification performance (on the Switchboard telephone conversation transcription task) to models estimated with the word-level MBR criterion. While this is reported purely as an observation, there exists some theoretical justification in favour of use of the MPE criterion over the word-level MBR criterion. This justification is presented in Chapter 7, as well as further arguments in favour of alternative formulations of the MBR criterion which incorporate knowledge of more detailed acoustic modelling errors.

¹Note that multiple correct phoneme-level transcriptions might exist corresponding to alternative word pronunciations. This issue is handled in Povey (2003).

3.3 Summary

This chapter has provided an overview of different model estimation paradigms. The use of discriminative learning with regard to acoustic model estimation has been motivated. The major discriminative acoustic model estimation techniques which have been employed to date have been introduced. While performance gains over standard ML-estimated models are reported for all of these discriminative training techniques, there is strong experimental evidence that MBR-estimated acoustic models exhibit superior performance to CML-estimated models for large vocabulary speech recognition tasks. Research into discriminative acoustic model estimation is ongoing and the optimal technique remains unknown. In the context of continuous speech recognition, use of the Levenshtein error metric to quantify misclassifications provides theoretical motivation for the use of the MBR method instead of MCE or maximum margin estimation.

The next chapter introduces the field of speaker adaptation and explains how discriminative learning approaches are motivated with regard to the speaker adaptation task. Subsequently, Chapter 5 details the implementation of MBR acoustic model estimation and adaptation. This provides the background for the main contributions of the thesis, detailed in subsequent chapters.

Chapter 4

Discriminative acoustic model adaptation

Chapter 3 has justified the use of discriminative learning for the task of acoustic model estimation in ASR systems. In this chapter, the goal of speaker adaptation is explained, previously introduced techniques designed to accomplish this goal are presented, the use of discriminative learning is motivated for this task, and previous work in this field is reviewed.

This chapter provides the background for the work presented in Chapter 8, which refines current unsupervised MBR-based acoustic model adaptation techniques to incorporate confidence information. In particular, the linear regression framework for acoustic model adaptation used in the experimental work of Chapter 8 is justified by comparing it with other approaches to the speaker adaptation task. Since generative (maximum likelihood) and discriminative (minimum Bayes risk) linear regression are compared experimentally in Chapter 8, these approaches are introduced in detail.

The chapter is structured as follows. Firstly, the speaker and environment adaptation tasks are introduced in Section 4.1. Feature-based adaptation is summarised in Section 4.2. A more detailed discussion of model-based adaptation is given in Section 4.3 to motivate the linear regression model adaptation framework used in the experimental work of this thesis. Discriminative model adaptation is introduced in Section 4.4, which also reviews previous work in this particular field. Section 4.5 provides a concluding discussion.

4.1 Speaker and environment adaptation

4.1.1 Speaker adaptation

Speaker dependent (SD) ASR systems are designed to recognise the speaker used to train the system. In contrast, speaker independent (SI) systems are designed to recognise any speaker. Given the same amount of training data, an SD system typically displays significantly better performance than an SI system (Woodland (2001)). This is due to a greater mismatch between the test data and the speech models in the case of the SI system. This mismatch is due to physiological and linguistic differences between speakers. Physiological

characteristics of a speaker include gender, age, health and vocal tract length and shape. Linguistic characteristics include dialect, accent, intonation, loudness and speaking rate.

Speaker adaptation is a set of techniques used by SI recognition systems to alter the system behaviour as it encounters new speakers or environmental conditions. If a large amount of speaker data is available (i.e. an amount comparable to the volume of data used to train the SI system), then standard model estimation techniques can be used to estimate an SD system using this data. However, such large quantities of data are often unavailable and the adaptation procedure must capitalise upon knowledge of both the SI system and a relatively small amount of speaker data, called the adaptation data, to estimate a speaker-adapted system. The adaptation procedure generally attempts to reduce the mismatch between the test data and the speech models. When the correct transcription of the adaptation data is available the task is referred to as supervised adaptation. Unsupervised adaptation adapts the system without knowledge of the correct transcription of the adaptation data.

4.1.2 Environment adaptation

Environment-independent ASR systems are designed to operate in changing acoustic environments. It should be noted that some of the techniques developed for the purpose of speaker adaptation can also be used to adapt such systems to reduce the level of mismatch between the models and the test data. This mismatch may be due to changing background noise conditions (e.g. car engine or office noise) or the effect of different acoustic channels (e.g. a microphone or a telephone line) upon the data.

As explained in Section 2.2, speech models comprise both language and acoustic models. Some attention has been paid to the task of language model adaptation (Gotoh and Renals (2000), Kuhn and De Mori (1990), Lau et al. (1993)), but the majority of research into speaker adaptation has focussed on acoustic adaptation. A diverse range of acoustic adaptation methods have been previously introduced. These techniques can be broadly categorized as either feature-based or model-based adaptation, as will now be explained.

4.2 Feature-based adaptation

Feature-based speaker adaptation, also referred to as feature normalisation, reduces inter-speaker, or inter-environment, acoustic variability purely via the adjustment of acoustic features. Vocal tract length normalization (VTLN) (Lee and Rose (1996)) is an example of a feature normalisation method. The VTLN method attempts to neutralise the effect of differing vocal tract lengths amongst training and test-set speakers via application of a scalar ‘warp factor’ to alter the centre frequencies of the filterbanks used in calculation of MFCC or PLP coefficients (see Section 2.1). This warp factor is chosen such that the likelihood of the models corresponding to the speaker data is maximised. Application of the appropriate warp factors to both the training and test data reduces the variability caused by vocal tract length differences and consequently reduces the mismatch between the estimated SI models and the test data.

Cepstral mean normalisation (CMN) (Atal (1974)) is another feature normalisation technique, used to compensate for different acoustic channel conditions. Using the CMN technique, the mean of a set of cepstral features associated with a particular channel (e.g. a particular microphone) is subtracted from each cepstral feature vector in the set. This subtraction removes, to some extent, the channel-specific properties of the cepstral speech representation. Similarly, cepstral variance normalisation (CVN) is often used in conjunction with CMN to reduce variance introduced by acoustic channel conditions.

The feature normalisation techniques described above have proved successful for tasks which involve adaptation to acoustic channels or speaker gender. They are often used in conjunction with model-based adaptation techniques.

4.3 Model-based adaptation

Model-based adaptation schemes use the adaptation data to re-estimate the parameters of the SI models, yielding an adapted system for recognition of a particular speaker or environment. Some successful methods include maximum a posteriori (MAP) estimation, maximum likelihood linear regression (MLLR) and speaker adaptive training (SAT), introduced in Gauvain and Lee (1994), Leggetter and Woodland (1995) and Anastasakos et al. (1996) respectively. These techniques prove useful for a reasonable volume of adaptation data, i.e. over 10 seconds of speaker or environment-specific speech. Experiments with small volumes of adaptation data in Leggetter and Woodland (1995) show that MLLR can be useful when using an average of approximately 11 seconds of speech per speaker. However, when using adaptation data volumes of the order of 2 or 3 seconds of speech, some alternative approaches to speaker adaptation are more successful. These approaches, sometimes called speaker clustering (Woodland (2001)) or speaker space (Kuhn et al. (2000)) methods, are introduced in Section 4.3.4. The MAP, MLLR and SAT techniques are discussed in Sections 4.3.1, 4.3.2 and 4.3.3 respectively.

4.3.1 MAP adaptation

Maximum a-posteriori (MAP) model estimation was introduced in Section 3.1.2. This theory was applied to the task of acoustic model adaptation in Gauvain and Lee (1994). The MAP estimate maximises the posterior probability of the model parameters θ , given the adaptation data \mathbf{o}_1^T and its associated transcription \hat{w}_1^M . This is expressed in Equation 4.3.1.

$$\begin{aligned}\theta_{\text{MAP}} &= \arg \max_{\theta} p(\theta | \mathbf{o}_1^T, \hat{w}_1^M) \\ &= \arg \max_{\theta} p(\mathbf{o}_1^T, \hat{w}_1^M | \theta) p(\theta)\end{aligned}\tag{4.3.1}$$

Note that in Equation 4.3.1, Bayes' rule has been used to reorganise the probabilities and the denominator $p(\mathbf{o}_1^T, \hat{w}_1^M)$ is discarded as it is independent of the model parameters θ . In Gauvain and Lee (1994), the form of the prior distribution $p(\theta)$ is specified for HMM

acoustic model parameters and the EM algorithm is applied to derive iterative parameter update equations.

Unlike maximum likelihood (ML) estimation, MAP estimation provides reliable model estimates with only a small amount of training data by incorporating knowledge of the prior distribution of model parameters into the estimation procedure. Such robustness with small amounts of data renders MAP estimation suitable for the speaker adaptation task. In Gauvain and Lee (1994), MAP-estimated SD models are shown to yield superior performance to ML-estimated SD models for the resource management medium vocabulary speech recognition task (Price et al. (1988)) for volumes of adaptation data less than 30 minutes.

One criticism of MAP adaptation is that the adaptation process is slow, i.e. it requires a relatively large amount of adaptation data to become effective. This is because only the model parameters corresponding to the adaptation data are re-estimated. Several methods have been proposed to overcome this limitation, notably predictive model adaptation (Cox (1995), Ahadi and Woodland (1997)) and structural MAP (SMAP) (Shinoda and Lee (2001)). These extensions to MAP expedite the adaptation process via knowledge of the relationships between the parameters of the acoustic model. These relationships are then used to adapt parameters which have a relatively low volume of associated adaptation data, thus accelerating the adaptation process. Similar to these extensions to MAP adaptation, linear regression speaker adaptation exploits knowledge of the relationships between acoustic model parameters. Since linear regression delivers superior performance with lower volumes of adaptation data, it is the technique chosen for the experimental work of this thesis, and a relatively detailed introduction to this method is now presented.

4.3.2 Maximum likelihood linear regression

Linear regression speaker adaptation (Leggetter and Woodland (1995)) uses adaptation data from a speaker to estimate one or more affine transforms of the speaker independent acoustic model parameters. In the case of continuous density HMMs with Gaussian mixture state output distributions of dimension n , the mean of a SI Gaussian mixture component $\boldsymbol{\mu}_s$ is transformed according to Equation 4.3.2.

$$\hat{\boldsymbol{\mu}}_s = \mathbf{A}\boldsymbol{\mu}_s + \mathbf{b} = \mathbf{W}\boldsymbol{\xi}_s \quad (4.3.2)$$

The symbol $\hat{\boldsymbol{\mu}}_s$ represents the adapted mean, $\mathbf{W} = [\mathbf{b} \ \mathbf{A}]$, $\boldsymbol{\xi}_s$ is the extended mean vector $[1 \ \boldsymbol{\mu}_s^\top]^\top$, \mathbf{A} is an $n \times n$ matrix and \mathbf{b} is an n -dimensional vector called the bias.

The affine transform described above is typically shared across many mixture components. This parameter-tying is implemented via a regression class tree (Leggetter and Woodland (1995), Young et al. (2003)). This structure groups the mixture components of the ASR system into hierarchically-defined clusters called regression classes as illustrated in Figure 4.1. Each node in the tree defines a regression class. Each mixture component in the system belongs to one and only one leaf node of the tree. These leaf nodes are called base regression classes. The leaf nodes are then tied together at parent nodes to define higher-level regression classes which comprise all of the components specified in their descendant

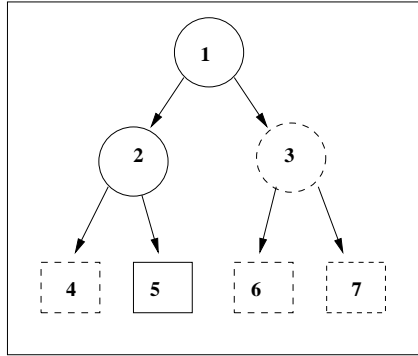


Figure 4.1: *Regression class tree. Squares represent base regression classes and circles denote higher-level regression classes. Dashed borders indicate regression classes whose associated data is insufficient for robust transform estimation.*

leaf nodes. The root node of the regression class tree therefore denotes a regression class which comprises all of the mixture components in the system.

In addition to defining how parameters are tied, the regression class tree is used to ensure that the generated transforms correspond to the volume and nature of the available adaptation data. This is done by specifying an occupancy threshold. The volume of adaptation data associated with the components of the regression class, i.e. the sum (over all time frames of adaptation data) of the occupancies of each of the components associated with the regression class, must exceed this occupancy threshold before a transform is generated for the regression class. An individual component is adapted using the transform corresponding to the parent regression class lowest in the regression class tree hierarchy for which the occupancy threshold is exceeded.

In Figure 4.1, nodes with dotted borders represent regression classes with insufficient occupancy. Components belonging to base class 4 are transformed according to the transform generated at node 2, and components belonging to base classes 6 and 7 are transformed according to the transform generated at the root node 1. Unlike MAP adaptation, components can be adapted if they have little or no associated adaptation data, provided their regression class as a whole has enough associated data. This renders linear regression adaptation superior to MAP adaptation at lower volumes of adaptation data. In the experimental comparison of the methods discussed in Chapter 9 of Huang et al. (2001), acoustic models adapted using linear regression yield superior performance to MAP-adapted models when the adaptation data comprises less than 400 utterances. The task used for the evaluation is a 60000-word dictated speech recognition application.

In the case of maximum likelihood linear regression (MLLR), transforms are chosen to maximise the likelihood of the models corresponding to the adaptation data. The re-estimation equation for the mean transform is derived using the expectation-maximisation framework in Leggetter and Woodland (1995). The transform estimate \mathbf{W} for the mean of

component s is shown to obey Equation 4.3.3.

$$\sum_{m \in \mathcal{R}(s)} \mathbf{C}_m^{-1} \gamma_m \mathbf{W} \boldsymbol{\xi}_m \boldsymbol{\xi}_m^\top = \sum_{m \in \mathcal{R}(s)} \mathbf{C}_m^{-1} \boldsymbol{\Gamma}_m \boldsymbol{\xi}_m^\top \quad (4.3.3)$$

The set $\mathcal{R}(s)$ comprises all mixture components m in the same regression class as component s , $\boldsymbol{\xi}_m$ is the extended mean of component m and \mathbf{C}_m is the covariance of component m . The quantity γ_m is the overall (summed over all training examples r and frames t) occupancy of component m , given by Equation 4.3.4.

$$\gamma_m = \sum_{r=1}^R \sum_{t=1}^{T(r)} \gamma_m(t | \hat{w}_1^{M(r)}, \mathbf{o}_1^{T(r)}, \theta) \quad (4.3.4)$$

The quantity $\boldsymbol{\Gamma}_m$ is defined by Equation 4.3.5, where $\mathbf{o}_t(r)$ is the t -th feature vector in the sequence $\mathbf{o}_1^{T(r)}$.

$$\boldsymbol{\Gamma}_m = \sum_{r=1}^R \sum_{t=1}^{T(r)} \gamma_m(t | \hat{w}_1^{M(r)}, \mathbf{o}_1^{T(r)}, \theta) \mathbf{o}_t(r) \quad (4.3.5)$$

Under the additional assumption that mixture distributions are modelled with diagonal covariances, it is shown in Leggetter and Woodland (1995) that the i -th row of the mean transform \mathbf{W} is given by $\mathbf{W}^{(i)}$ in Equation 4.3.6.

$$\mathbf{W}^{(i)} = \mathbf{G}^{(i)-1} \mathbf{k}^{(i)} \quad (4.3.6)$$

The quantities $\mathbf{G}^{(i)}$ and $\mathbf{k}^{(i)}$ are given by Equations 4.3.7 and 4.3.8 respectively.

$$\mathbf{G}^{(i)} = \sum_{m \in \mathcal{R}(s)} \frac{1}{\sigma_m^{2(i)}} \gamma_m \boldsymbol{\xi}_m \boldsymbol{\xi}_m^\top \quad (4.3.7)$$

$$\mathbf{k}^{(i)} = \sum_{m \in \mathcal{R}(s)} \frac{1}{\sigma_m^{2(i)}} \boldsymbol{\Gamma}_m^{(i)} \mu_m^{(i)} \boldsymbol{\xi}_m \quad (4.3.8)$$

In the above equations, $\sigma_m^{2(i)}$ is the variance of the i -th dimension of component m , $\mu_m^{(i)}$ is the i -th dimension of the mean of component m , and $\boldsymbol{\Gamma}_m^{(i)}$ is the i -th dimension of vector $\boldsymbol{\Gamma}_m$. In this section, only the theory of MLLR mean transformation has been presented. It should be noted that an extension of the MLLR formalism for transforming component covariances is presented in Gales and Woodland (1996). It should also be noted that the optimal technique for defining the regression class tree structure is unknown and some research in this area has been conducted (Gales (1996), Mandal et al. (2006)). However, grouping components whose means are close to each other in acoustic space is a simple and successful method (Leggetter and Woodland (1995)) and is therefore the technique used in the work of this thesis (Chapter 8).

4.3.3 Speaker Adaptive Training

The idea behind speaker adaptive training (SAT, Anastasakos et al. (1996)) is to decouple inter-speaker variance and phonetic variance when estimating an SI acoustic model. The SAT framework simultaneously estimates sets of speaker-dependent affine transforms of the acoustic models using MLLR (one set of transforms for each speaker in the training set) and a speaker independent ‘canonical’ model. The parameters of the canonical model are estimated with the speaker transforms applied to the model. These transforms account for much of the inter-speaker acoustic variance and consequently the canonical model displays less variance than a standard SI system. During recognition the canonical model is adapted to test speakers using MLLR. In Anastasakos et al. (1996), the SAT technique is evaluated on Wall Street Journal large vocabulary transcription tasks and yields significant classification performance improvements over standard MLLR.

4.3.4 Speaker space adaptation

The MAP and MLLR techniques do not explicitly use knowledge of how SD models are distributed in relation to each other. Adaptation approaches which take advantage of this knowledge are collectively called speaker space adaptation methods (Kuhn et al. (2000)). Speaker space adaptation involves positioning a new speaker in a low-dimensional model space and subsequently inferring the parameters of the high-dimensional SD model from this estimated position in the low-dimensional space. Gender-dependent modelling with a pre-recognition step which estimates the speaker gender is a simple example of speaker space adaptation. More complex speaker space methods include cluster adaptive training (CAT) (Gales (2000)) and eigenvoices (Kuhn et al. (2000)).

Both CAT and eigenvoices use the idea of basis models. In the case of CAT, each basis model represents a cluster of speakers. In the case of eigenvoices, the basis models represent the principal components of inter-speaker variability. Once the basis models have been estimated, they are used to represent a speaker space, in the sense that all new speakers adopt models which are a linear combination of the basis models. Equation 4.3.9 represents the CAT estimation of the mean of a particular Gaussian mixture component m of a new speaker model as a weighted sum of the basis model means, $\boldsymbol{\mu}_m^{(k)}$. The adaptation process involves only the estimation of a small number of weights λ_k . Typically only the mean parameters are adapted when using CAT or eigenvoices.

$$\boldsymbol{\mu}_m = \sum_k \lambda_k \boldsymbol{\mu}_m^{(k)} \quad (4.3.9)$$

With small quantities of adaptation data, CAT and eigenvoices display superior performance to MLLR due to the small amount of parameters involved in the adaptation process. In Gales (2000), for an internal IBM dictation task, it is found that CAT displays superior performance to MLLR when the number of adaptation utterances is less than 10. In Kuhn et al. (2000), for the ISOLET isolated letter recognition task (Cole et al. (1990)), the eigenvoice adaptation technique delivers superior performance to MLLR when the number of adaptation letters is less than 20.

4.3.5 Summary

Several different model adaptation techniques have now been presented. The effectiveness of each technique depends upon the amount of adaptation data available. MAP adaptation is the most flexible technique, in the sense that it uses no parameter tying, and consequently has the largest number of free parameters of all the adaptation techniques described. With the availability of a large amount of supervised adaptation data, MAP provides the best performance of the adaptation methods discussed. Given lower volumes of adaptation data, the parameter-tying mechanism used by MLLR adaptation becomes useful, and the performance of MLLR-adapted models is superior to that of MAP-adapted models. Given a very low volume of adaptation data, the number of free parameters associated with MLLR is too large, and the technique may overfit the adaptation data. Speaker space adaptation schemes are usually constrained to have fewer free parameters than MLLR. Since a small number of parameters can be robustly estimated with a small quantity of data, speaker space adaptation methods are more effective than MLLR with such amounts of adaptation data.

In the adaptation tasks described in Chapter 8, the volume of adaptation data is such that linear regression adaptation is the preferred technique. Therefore, the theoretical and experimental work of this thesis focusses on the use of linear regression adaptation. In particular, the use of discriminative learning in combination with linear regression is explored. The use of discriminative learning with regard to acoustic model adaptation in general is now introduced.

4.4 Discriminative speaker adaptation

The successful application of discriminative learning methods to the acoustic model estimation task, described in Chapter 3, has prompted interest in their application to acoustic model adaptation. The arguments presented in Chapter 3 in favour of discriminative learning for acoustic model estimation hold also for the case of acoustic model adaptation. Consequently, several of the discriminative approaches described in Chapter 3 have been applied to the model adaptation schemes described in Section 4.3. While some success has been reported when using discriminative approaches for the task of supervised acoustic model adaptation, there have been notably fewer reports of success in the unsupervised scenario. Therefore Chapter 8 concentrates upon unsupervised discriminative acoustic model adaptation. As stated beforehand, linear regression provides the most suitable adaptation framework, so focus is given to the application of discriminative learning to unsupervised linear regression adaptation. A review of previous work in the field of discriminative acoustic model adaptation is now presented.

4.4.1 Previous work

When initially proposed, the model adaptation methods described in Section 4.3 used generative learning techniques (either MAP or ML) to estimate their governing parameters. Using MAP or ML criteria, the acoustic model is adapted in such a way that the a-posteriori probability or likelihood of the acoustic models corresponding to the adaptation data is maximised. Discriminative acoustic model adaptation alters the acoustic models such that a discriminative criterion is optimised.

Discriminative versions of MAP (Povey et al. (2003a), Povey et al. (2003b)), linear regression (Gunawardana and Byrne (2001), Wu and Huo (2002), Wang and Woodland (2004)), speaker adaptive training (Tsakalidis et al. (2002), Wang and Woodland (2002)) and speaker space adaptation (Yu and Gales (2006)) have been previously proposed.

Discriminative MAP adaptation involves optimisation of a smoothed discriminative criterion. A prior distribution over the acoustic model parameters is added to the discriminative criterion to construct this smoothed criterion. An example of such a prior distribution is used in a method called I-smoothing, introduced in Section 5.2.4. Results reported in Povey et al. (2003b) show that use of the smoothed conditional ML (CML) criterion consistently yields a lower WER than standard MAP adaptation for the task of supervised adaptation of Switchboard-trained (Godfrey et al. (1992)) acoustic models to the Voicemail task (Padmanabhan et al. (1997)).

A discriminative version of SAT, where the CML criterion function is optimised instead of the ML criterion function, is introduced in Tsakalidis et al. (2002). Consistent WER improvements over ML-based SAT on Switchboard conversational speech recognition tasks (Godfrey et al. (1992)) are reported. In Wang and Woodland (2002), a version of SAT which replaces the ML criterion with the MPE criterion is presented. Some comparisons are made between CML-based SAT and MPE-based SAT for Switchboard speech recognition tasks, but the reported results show no significant performance difference between the techniques.

In the case of speaker space adaptation, a discriminative version of CAT employing the MPE criterion (Yu and Gales (2006)) has delivered significant classification performance improvement over ML-based CAT. This technique is also evaluated using Switchboard large vocabulary speech recognition tasks.

Note that all of the previous work discussed so far (discriminative MAP, SAT and CAT) has reported no results for the performance of the adaptation techniques in the unsupervised scenario. However, in the case of discriminative linear regression, some attempts have been made to apply the technique to the unsupervised task.

Discriminative linear regression

Discriminative linear regression adaptation deploys the linear regression adaptation framework described in Section 4.3.2. The difference between MLLR and discriminative linear regression is that the estimated affine transforms are chosen to optimise a discriminative criterion instead of the likelihood function. Discriminative LR variants corresponding to the CML criterion (Gunawardana and Byrne (2001)), the MCE criterion (Wu and Huo (2002)) and the MPE criterion (Wang and Woodland (2004)) have been introduced.

In Gunawardana (2001), the CML criterion is used to estimate the affine transforms of the linear regression speaker adaptation framework. This technique is called conditional MLLR (CMLLR). A comparison between CMLLR and MLLR is presented for a large vocabulary Switchboard speech recognition task in Gunawardana and Byrne (2001). The CMLLR method provides some small classification performance improvements over MLLR in the case of supervised adaptation. In the unsupervised case, when CMLLR is used after MLLR adaptation, small performance improvements over MLLR are recorded.

In Wu and Huo (2002) and He and Wu (2003), the MCE criterion is deployed within the linear regression adaptation framework. The resulting technique is called MCE linear regression (MCELR). In the experimental work of Wu and Huo (2002), supervised MCELR is compared with MLLR and shown to yield classification performance improvements for a syllable recognition task in Mandarin Chinese (Zu et al. (1996), Zu (1997)). However it should be noted that the MCELR systems use a greater number of transforms. It is therefore unclear whether this is a fair comparison. In He and Wu (2003), the performance of supervised MLLR and MCELR are compared for a Wall Street Journal transcription task, but again different node occupancies and therefore potentially a different number of transforms are used in each case.

In the experimental work of this thesis, some care is taken to ensure discriminative linear transforms use the same complexity control mechanism as MLLR transforms to achieve a fair comparison between the techniques. More details of complexity control in the case of discriminative linear regression are provided in Section 5.3.2.

In the case of minimum Bayes risk linear regression (MBRLR), the affine transforms are chosen to minimise the Bayes risk criterion. In Wang and Woodland (2004), MLLR and MBRLR are compared for the large vocabulary Switchboard and Wall Street Journal recognition tasks. Some small classification performance improvements are reported over MLLR in both the supervised and unsupervised MBRLR scenarios. It should be noted that the MPE formulation of the MBR criterion is used and that the MPE criterion is smoothed using a version of I-smoothing suitable for linear regression adaptation. Since the MBRLR technique is extended in Chapter 8, more details of the theory and implementation of MBRLR transform estimation, including the I-smoothing technique, are provided in Section 5.3.

4.5 Summary

This chapter has explained how acoustic model adaptation is used to improve the performance of speaker independent ASR systems. An overview of commonly used adaptation techniques has been provided. These techniques, when originally proposed, typically used generative learning principles to estimate their governing parameters. Discriminative versions of these adaptation methods select parameters which optimise a discriminative criterion. While some performance improvement has been reported when using discriminative speaker adaptation in a supervised scenario, very little significant performance improvement has been witnessed when the task is unsupervised. The work presented in Chapter

8 embraces the challenge of unsupervised discriminative acoustic model adaptation within the linear regression framework. Previous research into unsupervised MBRLR, summarised in this chapter, is extended by incorporating confidence information into the Bayes risk criterion.

Chapter 5

MBR theory and implementation

Previous chapters have motivated the use of MBR estimation and adaptation of acoustic models. The remainder of the thesis concentrates upon the application of the MBR technique to these tasks in the context of large vocabulary continuous ASR. The theory and implementation of MBR estimation and adaptation of HMM parameters are reviewed in this chapter.

The chapter is structured as follows. Section 5.1 reviews previous theoretical work on the minimisation of the MBR criterion and pursues Objective 1 of Section 1.5 by defining an auxiliary function to the MBR criterion. In Section 5.2, the implementation details of this optimisation procedure are explained in the context of large vocabulary continuous ASR systems. The theory and implementation of MBR linear regression adaptation are presented in Section 5.3. A summary of the arguments presented in the chapter is found in Section 5.4.

Note that Sections 5.2 and 5.3 discuss the approximations used in parameter optimisation as well as methods of improving the generalisation of the resulting acoustic models. Subsequent chapters will introduce alternative approximations and formulations of the MBR criterion. This chapter therefore provides a basis for the discussion of refinements and extensions introduced afterwards in the thesis.

5.1 MBR criterion optimisation

The MBR criterion has been introduced in Section 3.2.5 and the criterion is given by Equation 3.2.14. Adjustment of the acoustic model parameters such that the MBR criterion is minimised is generally performed by iterative updates of the model parameters. These updates are given by a set of equations known as the extended Baum-Welch (EBW) update formulae. Different versions of the EBW formulae have been introduced, corresponding to the CML, MCE and MBR criteria.

There have been several different justifications of the EBW update equations for continuous density HMMs. These justifications are briefly described in Sections 5.1.1, 5.1.2 and 5.1.3. As mentioned in relation to Objective 1 of the thesis in Section 1.5, these justifications

of the EBW update formulae either fail to guarantee the optimisation of the MBR criterion or fail to specify the learning rate D used in the EBW update formulae (see Equations 5.1.2 and 5.1.3). To address Objective 1 of the thesis, an auxiliary function for the MBR criterion is introduced in Section 5.1.4. As a by-product of the derivation of this auxiliary function, a theoretical value for the learning rate D is prescribed.

5.1.1 Discrete approximation of a continuous distribution

MBR estimation of discrete density HMM parameters was introduced in Na et al. (1995). A gradient descent technique called reduced gradient descent, applicable to parameters with constraints such as discrete probabilities, was used to estimate the model parameters. This initial work was advanced notably in two ways in Kaiser et al. (2002). Firstly, the theory was extended to the case of continuous speech recognition. Secondly, the MBR version of the EBW update equations for continuous density HMMs were derived. This derivation is in turn based upon the theory presented in Gopalakrishnan et al. (1989).

In Gopalakrishnan et al. (1989), it is shown that, for a criterion $R(\theta)$ of the form $P(\theta)/Q(\theta)$, where P and Q are polynomials with real coefficients in N variables, θ_i say, where the constraints

- $\theta_i \geq 0$ ($1 \leq i \leq N$) and
- $\sum_{i=1}^N \theta_i = 1$

are satisfied, then the following updates of the variables will result in an increase in the criterion for some positive real number C .

$$\hat{\theta}_i = \frac{\theta_i \left(\frac{\partial R(\theta)}{\partial \theta_i} + C \right)}{\sum_{k=1}^N \theta_k \frac{\partial R(\theta)}{\partial \theta_k} + C} \quad (5.1.1)$$

Notice that the negative of the MBR criterion can be expressed as a function of the above rational form. Further, in the case of HMMs with discrete density output distributions, the parameters θ_i of the output distributions are probabilities, and hence satisfy the constraints specified above. The parameter update equations of Equation 5.1.1 apply to all the acoustic parameters of discrete density HMMs: output distributions, mixture weights and transition probabilities. It is a simple exercise to differentiate the MBR criterion with respect to the model parameters.

It is additionally noted in Gopalakrishnan et al. (1989) that use of the finite positive C which guarantees an increase in the criterion for all parameter updates results in impractically slow convergence of the criterion to a local maximum. To speed up convergence, recommendations are provided for setting the constant C specifically for each set of parameters in one probability distribution. Note that use of this technique no longer guarantees convergence of the criterion. In practice, however, this is not found to be problematic.

Consider the case of HMMs with continuous density Gaussian mixture output distributions. The parameters of the output distributions (means and covariances) do not satisfy

the non-negativity and sum-to-one constraints specified above. A technique introduced in Normandin (1991) is used to handle this issue. It is shown that a continuous Gaussian distribution can be approximated using a discrete distribution with a finite number of parameters. This is done by partitioning the real line into a finite set of intervals. The probability associated with an observation x is then defined as the probability of the interval containing x , as specified by the Gaussian distribution (or zero if x belongs to the uppermost or lowermost interval of the real line). The update of a continuous-valued parameter, e.g. a Gaussian mean, can then be expressed in terms of the updated discrete parameters, in the limit as the interval size tends to zero and the number of intervals tends to infinity.

In Kaiser et al. (2002), this discrete approximation technique is reused to demonstrate that the MBR reestimation equations for means and covariances of diagonal covariance Gaussian mixture components are given, respectively, by Equations 5.1.2 and 5.1.3. These are the EBW update equations for continuous density HMMs in the case of the MBR criterion.

$$\hat{\boldsymbol{\mu}}_s = \frac{\sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} K(r, w_1^N | \theta) \sum_{t=1}^{T(r)} \gamma_s(t | w_1^N, \mathbf{o}_1^{T(r)}, \theta) \mathbf{o}_t(r) + D \boldsymbol{\mu}_s}{\sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} K(r, w_1^N | \theta) \sum_{t=1}^{T(r)} \gamma_s(t | w_1^N, \mathbf{o}_1^{T(r)}, \theta) + D} \quad (5.1.2)$$

$$\hat{\sigma}_s^{2(i)} = \frac{\sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} K(r, w_1^N | \theta) \sum_{t=1}^{T(r)} \gamma_s(t | w_1^N, \mathbf{o}_1^{T(r)}, \theta) (o_t^{(i)}(r))^2 + D(\sigma_s^{2(i)} + (\mu_s^{(i)})^2)}{\sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} K(r, w_1^N | \theta) \sum_{t=1}^{T(r)} \gamma_s(t | w_1^N, \mathbf{o}_1^{T(r)}, \theta) + D} - (\hat{\mu}_s^{(i)})^2 \quad (5.1.3)$$

In the above equations, $\hat{\boldsymbol{\mu}}_s$ and $\hat{\sigma}_s^{2(i)}$ are, respectively, the updated mean and updated variance of the i -th dimension of mixture component s , $\boldsymbol{\mu}_s$ and $\sigma_s^{2(i)}$ are, respectively, the current mean and current variance of the i -th dimension, $\hat{\mu}_s^{(i)}$ and $\mu_s^{(i)}$ are, respectively, the i -th dimension of the updated mean and current mean, $\mathbf{o}_t(r)$ is the acoustic feature vector of the t -th frame of the r -th training example and $o_t^{(i)}(r)$ is the i -th dimension of this vector. The set \mathcal{W} is the hypothesis space, $\gamma_s(t | w_1^N, \mathbf{o}_1^{T(r)}, \theta)$ is the occupancy of component s at time t given hypothesis w_1^N and observation sequence $\mathbf{o}_1^{T(r)}$ and $K(r, w_1^N | \theta)$ is the posterior-weighted relative error of hypothesis w_1^N as defined by Equation 5.1.4.

$$K(r, w_1^N | \theta) = P(w_1^N | \mathbf{o}_1^{T(r)}, \theta) [L_{\text{av}}(r) - L(\hat{w}_1^{M(r)}, w_1^N)] \quad (5.1.4)$$

Here $L(A, B)$ is the Levenshtein distance between sequences A and B , $\hat{w}_1^{M(r)}$ is the reference hypothesis, and $L_{\text{av}}(r)$ is the average error of all hypotheses, given by Equation 5.1.5.

$$L_{\text{av}}(r) = \sum_{w_1^N \in \mathcal{W}} P(w_1^N | \mathbf{o}_1^{T(r)}, \theta) L(\hat{w}_1^{M(r)}, w_1^N) \quad (5.1.5)$$

The value of the learning rate D is the limit of C (in Equation 5.1.1) as the approximating interval width (of the discrete approximation to the Gaussian distribution) tends to zero

and the number of intervals tends to infinity. Note that it is not certain that this limit exists, and hence the use of the EBW equations may not guarantee a decrease of the MBR criterion. However, in practice, these equations are found to be useful for iteratively re-estimating HMM parameters. That is, with an appropriate, empirically-defined choice of the constant D (as discussed in Section 5.2.3) the re-estimation equations yield a reduction in the MBR criterion with each iteration. An alternative derivation of the EBW update equations using a weak sense auxiliary function is explained below.

5.1.2 Weak sense auxiliary function

As explained in Section 3.1.3, when optimising the parameters of HMMs to maximise their likelihood, the EM algorithm can be deployed. This algorithm iteratively re-estimates the model parameters via maximisation of an appropriate auxiliary function¹. An increase in the auxiliary function guarantees that the model likelihood does not decrease. Indeed the EM algorithm will converge to either a local maximum or saddle point of the likelihood function.

In the case of discriminative estimation of HMM parameters, such useful auxiliary functions have proven elusive. The idea of a weak sense auxiliary function is introduced in Povey et al. (2003a) and Povey (2003). Let θ denote the model parameters, θ' represent the current model parameters and let $R(\theta)$ be a criterion we wish to maximise. Then a weak sense auxiliary function, $F(\theta|\theta')$, of the criterion $R(\theta)$ around θ' is a differentiable function of the parameters θ such that the gradient of $F(\theta|\theta')$ at θ' is the same as the gradient of $R(\theta)$ at θ' , as described by Equation 5.1.6.

$$\left. \frac{\partial R(\theta)}{\partial \theta} \right|_{\theta=\theta'} = \left. \frac{\partial F(\theta|\theta')}{\partial \theta} \right|_{\theta=\theta'} \quad (5.1.6)$$

If the function $F(\theta|\theta')$ is a concave function of the model parameters then it may be employed as an auxiliary function for maximising the criterion in an iterative EM-like algorithm. If a series of parameter updates yield a weak sense auxiliary function with zero gradient at θ' , then the criterion must also have zero gradient at θ' . If the updates made to the model parameters between each iteration are sufficiently small and aim to increase $F(\theta|\theta')$, then this point of zero gradient corresponds to a local maximum or saddle point of the criterion $R(\theta)$.

In Povey et al. (2003a), weak sense auxiliary functions are constructed for the CML and MPE criteria. A smoothing function, designed to introduce concavity to the weak sense auxiliary function, is added to each of these functions. This smoothing function is equal to the summation (over every mixture component s) of the log likelihood of the model for mixture s (mean $\boldsymbol{\mu}_s$ and covariance \mathbf{C}_s), given D_s datapoints with mean $\boldsymbol{\mu}'_s$ and covariance \mathbf{C}'_s (the current parameters). This smoothing function is expressed by Equation 5.1.7, where the sum is over each mixture component and the log likelihood term is given by Equation

¹This auxiliary function is the expected value of the logarithm of the model likelihood, where the probability distribution used to calculate the expected value uses the current estimate of the parameters.

5.1.8. The symbols θ and θ' represent the parameters and current parameters, respectively, and d is the dimension of the feature space.

$$Q_{\text{sm}}(\theta, \theta') = \sum_s Q(D_s, D_s \boldsymbol{\mu}'_s, D_s(\mathbf{C}'_s + \boldsymbol{\mu}'_s \boldsymbol{\mu}'_s{}^\top) | \boldsymbol{\mu}_s, \mathbf{C}_s) \quad (5.1.7)$$

$$\begin{aligned} Q(D_s, D_s \boldsymbol{\mu}'_s, D_s(\mathbf{C}'_s + \boldsymbol{\mu}'_s \boldsymbol{\mu}'_s{}^\top) | \boldsymbol{\mu}_s, \mathbf{C}_s) &= -\frac{D_s}{2} [d \log(2\pi) + \log \det \mathbf{C}_s + \text{tr}(\mathbf{C}_s^{-1} \mathbf{C}'_s) \\ &\quad + \text{tr}(\mathbf{C}_s^{-1} \boldsymbol{\mu}'_s \boldsymbol{\mu}'_s{}^\top) - 2\boldsymbol{\mu}'_s{}^\top \mathbf{C}_s^{-1} \boldsymbol{\mu}_s + \boldsymbol{\mu}_s{}^\top \mathbf{C}_s^{-1} \boldsymbol{\mu}_s] \end{aligned} \quad (5.1.8)$$

Such a smoothing function is globally maximal around the current parameter values and the configurable parameters D_s assume the role of individual learning rates for each mixture component.

In Povey (2003), it is shown that maximisation of this weak sense auxiliary function for the MBR criterion results in the EBW update equations 5.1.2 and 5.1.3. The mixture-specific learning rates D_s replace the constant D in the derivation of Section 5.1.1.

5.1.3 Extended Baum-Welch updates for general functions

The justifications of the EBW update equations described in Section 5.1.1 and 5.1.2 provide no guarantee that the MBR criterion will decrease with each iteration of parameter re-estimation. An alternative approach, introduced in Kanevsky (2004), shows that this is indeed the case provided the constant D is sufficiently large. It is shown that the general form of the EBW equations results in parameter updates which move the criterion in the desired direction given a sufficiently large D . This result is applicable not only to the MBR criterion but also to the CML and MCE criteria. While this work proves the existence of such a constant D , it does not prescribe a value for this constant.

5.1.4 Auxiliary function for the MBR criterion

One of the objectives of this thesis (Objective 1 of Section 1.5) is to investigate if an auxiliary function can be specified which both justifies the MBR EBW parameter update equations and specifies the learning rate used in these equations.

Applying similar arguments to those used in Gunawardana (2001) and later in Axelrod et al. (2007) in the case of the CML criterion, an auxiliary function for the MBR criterion can be established and consequently the EBW parameter update equations derived from this auxiliary function. Importantly, a theoretical value for the learning rate D is prescribed as a by-product of this theory. For the sake of continuity, the details of this thesis contribution are deferred to Appendix A. A brief outline of the results is given here.

Let θ and θ' be elements of the parameter space describing the means and covariances of the mixture components in a continuous density HMM. Let \mathcal{S} represent the set of possible hidden Markov model state sequences S associated with the training observation sequences.

Note that \mathcal{S} is the Cartesian product of state sequence spaces $\mathcal{S}_1 \times \mathcal{S}_2 \times \dots \times \mathcal{S}_r \times \dots \times \mathcal{S}_R$ where \mathcal{S}_r is the set of state sequences of length equal to the length of the r -th training observation sequence, $T(r)$. Here a state sequence s_1^T is the catenation of R sequences, $s_1^{T(1)} s_1^{T(2)} \dots s_1^{T(r)} \dots s_1^{T(R)}$, where the index corresponds to the training example index. Let \mathbf{o}_1^T represent the catenation of observation sequences of length corresponding the training utterances $\mathbf{o}_1^{T(1)} \mathbf{o}_1^{T(2)} \dots \mathbf{o}_1^{T(R)}$.

The function $F_{\text{MBR}}(\theta, \theta', D')$ is firstly defined by Equation 5.1.9.

$$F_{\text{MBR}}(\theta, \theta', D') = \int_{\mathbf{o}_1^T} \sum_{s_1^T \in \mathcal{S}} f(\mathbf{o}_1^T, s_1^T, \theta' | \theta') \log f(\mathbf{o}_1^T, s_1^T, \theta | \theta') d\mathbf{o}_1^T \quad (5.1.9)$$

Here $f(\mathbf{o}_1^T, s_1^T, \theta | \theta')$ is defined by Equation 5.1.10 and D' is defined by Equation 5.1.11, where $d(s_1^T)$ is a positive real function of the state sequence s_1^T and ϵ is a positive real.

$$f(\mathbf{o}_1^T, s_1^T, \theta | \theta') = p(\mathbf{o}_1^T, s_1^T | \theta) \left[\sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} 1_{\hat{\mathbf{o}}_1^T}(\mathbf{o}_1^T) a(w_1^N, \hat{w}_1^{M(r)}, \hat{\mathbf{o}}_1^{T(r)}, s_1^{T(r)}, \theta') + \frac{d(s_1^T)}{\epsilon} \right] \quad (5.1.10)$$

$$D' = \frac{1}{\epsilon} \int_{\mathbf{o}_1^T} \sum_{s_1^T \in \mathcal{S}} p(\mathbf{o}_1^T, s_1^T | \theta) d(s_1^T) d\mathbf{o}_1^T = \frac{1}{\epsilon} \sum_{s_1^T \in \mathcal{S}} p(s_1^T) d(s_1^T) \quad (5.1.11)$$

Above $1_{\hat{\mathbf{o}}_1^T}(\mathbf{o}_1^T)$ is an indicator function, and $\hat{\mathbf{o}}_1^T$ is the catenation of the observation sequences corresponding the training utterances, $\hat{\mathbf{o}}_1^{T(1)} \hat{\mathbf{o}}_1^{T(2)} \dots \hat{\mathbf{o}}_1^{T(R)}$. The symbol $\hat{\mathbf{o}}_1^{T(r)}$ represents the r -th training set observation sequence, $\hat{w}_1^{M(r)}$ is the corresponding transcription, and \mathcal{W} is the set of all possible word sequences.

The quantity $a(w_1^N, \hat{w}_1^{M(r)}, \hat{\mathbf{o}}_1^{T(r)}, s_1^{T(r)}, \theta')$ is defined in Equation 5.1.12.

$$a(w_1^N, \hat{w}_1^{M(r)}, \hat{\mathbf{o}}_1^{T(r)}, s_1^{T(r)}, \theta') = L(w_1^N, \hat{w}_1^{M(r)}) \left[p(w_1^N | \hat{\mathbf{o}}_1^{T(r)}, \theta') - p(w_1^N | s_1^{T(r)}) \right] \quad (5.1.12)$$

It is shown in Appendix A.3 that, under the assumption that \mathcal{W} is a finite set, if $0 < \epsilon \leq \epsilon_1$, then the function $F_{\text{MBR}}(\theta, \theta', D')$ is a valid auxiliary function for the MBR criterion $R_{\text{MBR}}(\theta)$. The quantity ϵ_1 is given by Equation 5.1.13.

$$\epsilon_1 = \frac{d_{\min} C_{\min}}{R \|\mathcal{W}\| L_{\max}} \quad (5.1.13)$$

The quantity d_{\min} is given by Equation 5.1.14, while L_{\max} and C_{\min} are given by Equations A.3.41 and A.3.42 respectively.

$$d_{\min} = \min_{s_1^T \in \mathcal{S}} \{d(s_1^T)\} \quad (5.1.14)$$

Using this auxiliary function, it is shown in Section A.5 that use of a mixture component specific learning rate constant D_s may be used in place of the constant D Equation 5.1.2.

Using the upper bound upon the quantity ϵ , it is shown that any D_s which satisfies the inequality of Equation 5.1.15 guarantees that the MBR criterion does not increase after the EBW parameter update.

$$D_s \geq \frac{R\|\mathcal{W}\|L_{max}}{C_{min}p(\hat{\mathbf{o}}_1^T|\theta')} \sum_{s_1^T \in \mathcal{S}} \sum_{t=1}^T 1_s(s_t)p(s_1^T) \quad (5.1.15)$$

5.2 Implementation of MBR parameter updates

When using small vocabulary systems it is possible to calculate the statistics required to perform the parameter updates specified by Equations 5.1.2 and 5.1.3 without approximation. However, in the context of large vocabulary continuous speech recognition, a prohibitively large amount of computation is required to gather these statistics. This is due to the size of the hypothesis space \mathcal{W} . A practical solution to this problem is to approximate the hypothesis space, and hence the resulting statistics, using an N-best list of the hypotheses of highest posterior (Kaiser et al. (2002)). In the context of large vocabulary continuous speech recognition, use of a word lattice (Woodland and Povey (2002)) to represent the hypothesis space is favoured because it is a more compact representation of such a list. For this reason, a lattice representation of the hypothesis space is used in the work of this thesis. The resulting implementation of MBR criterion optimisation is called lattice-based MBR.

5.2.1 Lattice-based MBR

Lattice-based MBR is introduced in Povey and Woodland (2002). Word lattices which include temporal alignment information, i.e. word start and end times, are used, and the lattice encodes the alignments of the acoustic data of highest posterior (Young et al. (2003)). A lattice is generated via a recognition pass of a speech utterance. Additionally, the alignments of the correct word sequence of highest posterior, generated using a constrained recognition pass, are added to the lattice produced by recognition. The resulting lattice represents a set of alternative word-level alignments of the acoustic data associated with an utterance.

The idea behind lattice-based MBR is not only to use the lattice as an approximation to the hypothesis space, but also to use the alignment information which is present in the lattice to save computation. Note that Equation 5.1.2 can be re-phrased as a sum over all possible alignments of the acoustic data as in Equation 5.2.1.

$$\hat{\boldsymbol{\mu}}_s = \frac{\sum_{r=1}^R \sum_{z \in \mathcal{Z}_r} K(r, z|\theta) \sum_{t=1}^{T(r)} \gamma_s(t|z, \mathbf{o}_1^{T(r)}, \theta) \mathbf{o}_t(r) + D\boldsymbol{\mu}_s}{\sum_{r=1}^R \sum_{z \in \mathcal{Z}_r} K(r, z|\theta) \sum_{t=1}^{T(r)} \gamma_s(t|z, \mathbf{o}_1^{T(r)}, \theta) + D} \quad (5.2.1)$$

The set \mathcal{Z}_r comprises all possible alignments of the utterance $\mathbf{o}_1^{T(r)}$ and $K(r, z|\theta)$ is given by Equation 5.2.2.

$$K(r, z|\theta) = P(z|\mathbf{o}_1^{T(r)}, \theta) [L_{av}(r) - L(\hat{w}_1^{M(r)}, w_z)] \quad (5.2.2)$$

In the above equation, w_z is the hypothesis associated with alignment z . Notice also that the average error $L_{\text{av}}(r)$ may also be expressed as a sum over alignments as in Equation 5.2.3.

$$L_{\text{av}}(r) = \sum_{z \in \mathcal{Z}_r} P(z | \mathbf{o}_1^{T(r)}, \theta) L(\hat{w}_1^{M(r)}, w_z). \quad (5.2.3)$$

Substituting the set of all possible alignments \mathcal{Z}_r with the set of alignments specified by the lattice, Equation 5.2.1 yields a practical approximation to the MBR mean update. Further, since an alignment is a sequence of lattice arcs, Equation 5.2.1 can be expressed in terms of lattice arcs as in Equation 5.2.4.

$$\hat{\boldsymbol{\mu}}_s = \frac{\sum_{r=1}^R \sum_{a \in \mathcal{A}_r} K(r, a | \theta) \sum_{t=a_{\text{start}}}^{a_{\text{end}}} \gamma_s(t | a, \mathbf{o}_1^{T(r)}, \theta) \mathbf{o}_t(r) + D \boldsymbol{\mu}_s}{\sum_{r=1}^R \sum_{a \in \mathcal{A}_r} K(r, a | \theta) \sum_{t=a_{\text{start}}}^{a_{\text{end}}} \gamma_s(t | a, \mathbf{o}_1^{T(r)}, \theta) + D} \quad (5.2.4)$$

The symbol a represents a lattice arc which in turn represents a word, its start time a_{start} and end time a_{end} . The set \mathcal{A}_r contains all arcs in the lattice and $K(r, a | \theta)$ is expressed by Equation 5.2.5.

$$K(r, a | \theta) = p(a | \mathbf{o}_1^{T(r)}, \theta) [L_{\text{av}}(r) - L(\hat{w}_1^{M(r)}, a)] \quad (5.2.5)$$

In the above equation, $p(a | \mathbf{o}_1^{T(r)}, \theta)$ is the posterior probability that arc a is included in any path, i.e. any contiguous sequence of arcs from the lattice start node to the lattice end node. The quantity $L(\hat{w}_1^{M(r)}, a)$ is the posterior-weighted sum of the Levenshtein error of all the lattice paths which include arc a , while $L_{\text{av}}(r)$ is the posterior-weighted sum of the Levenshtein error of all the lattice paths.

Calculation of the Levenshtein distance between a path in the lattice and the reference word sequence $\hat{w}_1^{M(r)}$ is non-trivial. This involves a dynamic programming alignment of the word sequence associated with the path and the reference word sequence. Since a lattice encodes many such paths, calculation of the quantities $L(\hat{w}_1^{M(r)}, a)$ and $L_{\text{av}}(r)$ becomes computationally expensive. Chapter 6 details techniques used to avoid such costly computation by approximating the Levenshtein distance between a lattice path and the reference transcription. These approximations assign an error $l(a)$ to each lattice arc a such that the overall error of each path is the sum of the errors associated with its composite arcs.

It should be noted that while this section has discussed only the approximations used in lattice-based MBR with regard to the mean update, identical approximations are used to implement the covariance update. There now follows an explanation of how the quantities $\gamma_s(t | a, \mathbf{o}_1^{T(r)}, \theta)$ and $K(r, a | \theta)$ of Equation 5.2.4 are calculated for each arc a .

5.2.2 Forward-backward algorithms

The mixture component occupancies $\gamma_s(t | a, \mathbf{o}_1^{T(r)}, \theta)$ of Equation 5.2.4 are calculated via a forward-backward pass over the models defined by each lattice arc a using the segment of acoustic data aligned with arc a . This is a standard forward-backward procedure, as implemented in the Baum-Welch algorithm described in Section 3.1.3. In order to calculate the quantities $K(r, a | \theta)$ for each arc a , a single lattice-level forward-backward pass suffices

when an error $l(a)$ is assigned to each lattice arc as described above. This forward-backward algorithm is introduced in Povey (2003) and detailed in Algorithm 1.

5.2.3 Learning rate D

The choice of an appropriate learning rate D in Equations 5.1.2 and 5.1.3 has mainly been studied in the context of the CML discriminative criterion. The successful policies for choosing D in the case of the CML criterion have been adopted for use in the MBR update equations. These approaches are summarised in this section.

A practical approach to the problem is taken in Valtchev et al. (1997), where D is twice the the minimum value which ensures that all variances are positive after the EBW variance update. This is possible by solving a quadratic equation in D for each Gaussian component and each dimension within that component. The value of D is subsequently set to a non-negative value which is at least twice the value which guarantees a positive variance update for each component and each dimension. It is noted, however, that use of this global learning rate leads to slow convergence of the criterion and an HMM-specific D (twice the minimum value which ensures a positive variance over all dimensions and components within the HMM) is shown to increase the rate of convergence.

In Povey (2003), each mixture component s is assigned its own learning rate D_s . Firstly the quantity D_s^{\min} is calculated, the value which guarantees a positive variance update for each dimension of component s . Furthermore a dependency upon the CML denominator occupancy (i.e. $\sum_{w_1^N \in \mathcal{W}} \sum_{r=1}^R \sum_{t=1}^{T(r)} \gamma_s(t | \mathbf{o}_1^{T(r)}, w_1^N, \theta)$), γ_s^{den} say, of component s is introduced as follows. For each mixture component s , D_s is set to $\max\{2D_s^{\min}, E\gamma_s^{\text{den}}\}$, where E is an additional constant.

This occupancy-dependent scheme for determining the learning rate in the case of CML training is adopted for MBR training in Povey (2003). The procedure is as follows. For each mixture component s :

1. Calculate D_s^{\min} , the minimum D required to ensure all variance updates are positive for component s .
2. Set $\gamma_s^{\text{den}} = \sum_r \sum_{a \in \mathcal{A}_r^{\text{den}}} K(r, a | \theta) \sum_{t=a_{\text{start}}}^{a_{\text{end}}} \gamma_s(t | a, \mathbf{o}_1^{T(r)}, \theta)$ where $\mathcal{A}_r^{\text{den}}$ denotes the subset of lattice arcs in the set of all lattice arcs \mathcal{A}_r for which $K(r, a | \theta)$ is negative. The symbols a_{start} and a_{end} denote the start and end time of arc a , respectively.
3. Set the learning rate D_s to $\max\{2D_s^{\min}, E\gamma_s^{\text{den}}\}$ where E is a configurable constant which typically assumes a value in the interval $[1, 2]$.

Since it has been shown to yield reasonably quick convergence of the MBR criterion, the above procedure is used to calculate the learning rate in the experimental work of this thesis. This concludes the discussion of the optimisation of the MBR criterion. Techniques used to improve the generalisation of MBR-estimated acoustic models are now discussed in Sections 5.2.4 and 5.2.5.

Algorithm 1 Lattice forward-backward

Let k index a partial ordering of the N arcs $a(k)$ in a lattice. This means that for any arc $a(k)$ and any $i > 0$, $a(k+i)$ does not appear before $a(k)$ in any left-to-right path in the lattice.

The sets $\text{pred}(k)$ and $\text{foll}(k)$ comprise, respectively, the immediate predecessor arcs of $a(k)$ and the immediately following arcs of $a(k)$. \mathcal{N} denotes the arcs linked to the lattice start node.

$l(k)$ is the error assigned to arc $a(k)$ and w_k is the word associated with arc $a(k)$.

s_k and e_k are, respectively, the start and end frames of arc $a(k)$.

$\mathbf{o}_{s_k}^{e_k}(r)$ represents the segment of acoustic data aligned to arc $a(k)$ and $p(\mathbf{o}_{s_k}^{e_k}(r), w_k|\theta)$ is the joint acoustic and language model probability of arc $a(k)$.

Forward pass

for $k = 1$ to N **do**

if $a(k)$ is linked to the lattice start node **then**

$$\alpha_k = p(\mathbf{o}_{s_k}^{e_k}(r), w_k|\theta)$$

$$\alpha'_k = l(k)$$

else

$$\alpha_k = p(\mathbf{o}_{s_k}^{e_k}(r), w_k|\theta) \sum_{a \in \text{pred}(k)} \alpha_a$$

$$\alpha'_k = l(k) + \frac{\sum_{a \in \text{pred}(k)} \alpha_a \alpha'_a}{\sum_{a \in \text{pred}(k)} \alpha_a}$$

end if

end for

Backward pass

for $k = N$ to 1 **do**

if $a(k)$ is linked to the lattice end node **then**

$$\beta_k = 1$$

$$\beta'_k = 0$$

else

$$\beta_k = \sum_{a \in \text{foll}(k)} p(\mathbf{o}_{s_a}^{e_a}(r), w_a|\theta) \beta_a$$

$$\beta'_k = \frac{\sum_{a \in \text{foll}(k)} \beta_a (\beta'_a + l(a))}{\sum_{a \in \text{foll}(k)} \beta_a}$$

end if

end for

$$L_{\text{av}}(r) = \frac{\sum_{a \in \mathcal{N}} \beta_a (\beta'_a + l(a))}{\sum_{a \in \mathcal{N}} \beta_a}$$

$$x = \sum_{a \in \mathcal{N}} p(\mathbf{o}_{s_a}^{e_a}(r), w_a|\theta) \beta_a$$

for all arcs a **do**

$$p(a|\mathbf{o}_1^{T(r)}, \theta) = \frac{\alpha_a \beta_a}{x}$$

$$L(\hat{w}_1^{M(r)}, a) = \alpha'_a + \beta'_a$$

end for

5.2.4 I-smoothing

Model parameters which optimise the MBR criterion often overfit the training data. The I-smoothing technique (Povey and Woodland (2002), Povey (2003)) is used to smooth the MBR criterion and alleviate this undesirable overfitting. In the case of the MBR criterion, a prior probability $\log p(\theta)$ over the acoustic parameters is subtracted from the criterion function. The smoothed MBR criterion $R_{\text{MBR}}^S(\theta)$ is given by Equation 5.2.6 (c.f. Equation 3.2.14).

$$R_{\text{MBR}}^S(\theta) = \frac{1}{R} \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} P(w_1^N | \mathbf{o}_1^{T(r)}, \theta) L(w_1^N, \hat{w}_1^{M(r)}) - \log p(\theta) \quad (5.2.6)$$

The prior distribution $p(\theta)$ is a product of joint prior distributions over the mean and covariance of each mixture component s in the system as expressed by Equation 5.2.7.

$$\log p(\theta) = \sum_s \log p(\boldsymbol{\mu}_s, \mathbf{C}_s) \quad (5.2.7)$$

The joint prior of the mean $\boldsymbol{\mu}_s$ and covariance \mathbf{C}_s is defined using the model likelihood, given τ^I data points of mean $\boldsymbol{\mu}_p$ and covariance \mathbf{C}_p (to be defined shortly). The model has mean $\boldsymbol{\mu}_s$ and covariance \mathbf{C}_s , and the likelihood is defined by Equation 5.2.8.

$$\log p(\boldsymbol{\mu}_s, \mathbf{C}_s) = Q(\tau^I, \tau^I \boldsymbol{\mu}_p, \tau^I (\mathbf{C}_p + \boldsymbol{\mu}_p \boldsymbol{\mu}_p^T) | \boldsymbol{\mu}_s, \mathbf{C}_s) + k \quad (5.2.8)$$

Here k is a normalisation term which ensures the probability distribution sums to one. The quantity Q is the log likelihood of the models given the data points, as expressed by Equation 5.1.8 in the general case, and by Equation 5.2.9 in the case of diagonal covariance matrices.

$$\begin{aligned} Q(\tau^I, \tau^I \boldsymbol{\mu}_p, \tau^I (\mathbf{C}_p + \boldsymbol{\mu}_p \boldsymbol{\mu}_p^T) | \boldsymbol{\mu}_s, \mathbf{C}_s) &= -\frac{\tau^I}{2} \left[\sum_{i=1}^d \log(2\pi\sigma_s^{2(i)}) \right. \\ &\quad \left. + \sum_{i=1}^d \frac{(\sigma_p^{2(i)} + \mu_p^{(i)2}) - 2\mu_p^{(i)} \mu_s^{(i)} + \mu_s^{(i)2}}{\sigma_s^{2(i)}} \right] \end{aligned} \quad (5.2.9)$$

In Equation 5.2.9, d is the number of dimensions of the acoustic feature vector and i indexes each dimension. The term τ^I affects the narrowness of the resulting prior distribution. High τ^I values reflect high confidence in the prior parameters. The family of prior distributions for I-smoothing differs from the family defined in Gauvain and Lee (1994) in the case of MAP parameter estimation. However use of the I-smoothing prior leads to a straightforward refinement to the EBW update equations.

When using I-smoothing, the maximum likelihood estimate of the means and covariances are used to define the mode of the prior distribution i.e. the mean $\boldsymbol{\mu}_p$ and covariance \mathbf{C}_p are defined as the ML estimates of the the mean and covariance respectively. Thus Equation 5.2.10 holds.

$$\log p(\boldsymbol{\mu}_s, \mathbf{C}_s) = Q(\tau^I, \tau^I \frac{\boldsymbol{\theta}_s^{ml}}{\gamma_s^{ml}}, \tau^I \frac{\boldsymbol{\Gamma}_s^{ml}}{\gamma_s^{ml}} | \boldsymbol{\mu}_s, \mathbf{C}_s) + k \quad (5.2.10)$$

The quantities γ_s^{ml} , θ_s^{ml} and Γ_s^{ml} are given by Equations 5.2.11, 5.2.12 and 5.2.13 respectively.

$$\gamma_s^{ml} = \sum_{r=1}^R \sum_{t=1}^{T(r)} \gamma_s(t | \hat{w}_1^{M(r)}, \mathbf{o}_1^{T(r)}, \theta) \quad (5.2.11)$$

$$\theta_s^{ml} = \sum_{r=1}^R \sum_{t=1}^{T(r)} \gamma_s(t | \hat{w}_1^{M(r)}, \mathbf{o}_1^{T(r)}, \theta) \mathbf{o}_t(r) \quad (5.2.12)$$

$$\Gamma_s^{ml} = \sum_{r=1}^R \sum_{t=1}^{T(r)} \gamma_s(t | \hat{w}_1^{M(r)}, \mathbf{o}_1^{T(r)}, \theta) \mathbf{o}_t(r) \mathbf{o}_t(r)^T \quad (5.2.13)$$

The I-smoothed MBR criterion (Equation 5.2.6) is optimised using a weak sense auxiliary function in Povey (2003). It is shown that, using the prior defined above, the adjusted EBW formulae shown in Equations 5.2.14 and 5.2.15 (in the case of diagonal covariances) can be used to optimise the I-smoothed MBR criterion.

$$\hat{\mu}_s = \frac{\sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} K(r, w_1^N | \theta) \sum_{t=1}^{T(r)} \gamma_s(t | w_1^N, \mathbf{o}_1^{T(r)}, \theta) \mathbf{o}_t(r) + \frac{\tau^I}{\gamma_s^{ml}} \theta_s^{ml} + D \mu_s}{\sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} K(r, w_1^N | \theta) \sum_{t=1}^{T(r)} \gamma_s(t | w_1^N, \mathbf{o}_1^{T(r)}, \theta) + \tau^I + D} \quad (5.2.14)$$

$$\begin{aligned} & \hat{\sigma}_s^{2(i)} \\ = & \frac{\sum_{r=1}^R \sum_{w_1^N} K(r, w_1^N | \theta) \sum_{t=1}^{T(r)} \gamma_s(t | w_1^N, \mathbf{o}_1^{T(r)}, \theta) (o_r^{t(i)})^2 + \frac{\tau^I}{\gamma_s^{ml}} \Gamma_s^{ml(i)} + D(\sigma_s^{2(i)} + (\mu_s^{(i)})^2)}{\sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} K(r, w_1^N | \theta) \sum_{t=1}^{T(r)} \gamma_s(t | w_1^N, \mathbf{o}_1^{T(r)}, \theta) + \tau^I + D} \\ & - (\hat{\mu}_s^{(i)})^2 \quad (5.2.15) \end{aligned}$$

In Equation 5.2.15 the superscript (i) denotes the dimension. The I-smoothing technique is used in the experimental work of Chapters 6 and 7.

5.2.5 Acoustic scaling and language model specificity

Due to the assumptions of HMM acoustic models (see Section 2.3), the dynamic range of acoustic model probabilities is typically much larger than that of LM probabilities. When combining acoustic and LM probabilities, for example during the Viterbi search algorithm (see Section 2.5), this mismatch is usually addressed by scaling the logarithm of the LM probabilities by an appropriate factor, known as the LM scaling factor. This method is known as LM scaling. An alternative to LM scaling is to scale acoustic model probabilities by the inverse of the LM scaling factor. In the case of the Viterbi algorithm, both techniques are equivalent, i.e. the state sequence of maximal posterior probability is independent of the scaling method used. However it has been noted (Woodland and Povey (2002)) that use of LM scaling for the purposes of calculation of the posterior probabilities of a set of state sequences results in a posterior probability distribution which is sharply-peaked at the state sequence of maximal posterior probability. When using acoustic scaling, this

posterior probability distribution is broader (i.e. competing state sequences have larger posterior probabilities). It has additionally been observed (Woodland and Povey (2002)) that use of acoustic scaling leads to improved generalisation of discriminatively-estimated acoustic models.

Similarly, the use of lower-order LMs during training, for example a zero-gram or uni-gram, is another method of generating stronger (i.e. of higher posterior probability) competing hypotheses. An investigation into the relationship between LM usage and discriminative training is conducted in Schluter et al. (1999). It is concluded that use of a unigram LM during training generally yields acoustic models which generalise better than those estimated using LMs of lower or higher order. Acoustic scaling and unigram LMs are deployed in the experimental work of Chapters 6 and 7.

5.3 Minimum Bayes risk linear regression

In the case of MBR linear regression, the re-estimation equations for both the mean and diagonal variance transforms are derived in Wang and Woodland (2004) using a weak sense auxiliary function. Assuming that the HMM state output distributions are Gaussian mixtures modelled with diagonal covariances, the i -th row of the mean transform \mathbf{W} of mixture component s for a particular speaker is given by Equation 5.3.1.

$$\mathbf{W}^{(i)} = \mathbf{G}_{\text{MBR}}^{(i)-1} \mathbf{k}_{\text{MBR}}^{(i)} \quad (5.3.1)$$

The matrix $\mathbf{G}_{\text{MBR}}^{(i)}$ and the vector $\mathbf{k}_{\text{MBR}}^{(i)}$ are defined by Equations 5.3.2 and 5.3.3 respectively.

$$\mathbf{G}_{\text{MBR}}^{(i)} = \sum_{m \in \mathcal{R}(s)} \frac{1}{\sigma_m^{2(i)}} (\gamma_m^{\text{MBR}} + D_m) \boldsymbol{\xi}_m \boldsymbol{\xi}_m^{\text{T}} \quad (5.3.2)$$

$$\mathbf{k}_{\text{MBR}}^{(i)} = \sum_{m \in \mathcal{R}(s)} \frac{1}{\sigma_m^{2(i)}} (\theta_m^{\text{MBR}(i)} + D_m \mu_m^{(i)}) \boldsymbol{\xi}_m \quad (5.3.3)$$

In Equations 5.3.2 and 5.3.3, $\mathcal{R}(s)$ is the regression class containing component s , D_m is the learning rate discussed in Section 5.2.3, $\mu_m^{(i)}$ is the i -th dimension of the mean of component m , $\boldsymbol{\xi}_m$ is the extended mean vector $[1 \quad \boldsymbol{\mu}_m^{\text{T}}]^{\text{T}}$, $\sigma_m^{2(i)}$ is the variance of the i -th dimension of component m , γ_m^{MBR} is described by Equation 5.3.4 and $\theta_m^{\text{MBR}(i)}$ is given by Equation 5.3.5.

$$\gamma_m^{\text{MBR}} = \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} K(r, w_1^N | \theta) \sum_{t=1}^{T(r)} \gamma_m(t | w_1^N, \mathbf{o}_1^{T(r)}, \theta) \quad (5.3.4)$$

$$\theta_m^{\text{MBR}(i)} = \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} K(r, w_1^N | \theta) \sum_{t=1}^{T(r)} \gamma_m(t | w_1^N, \mathbf{o}_1^{T(r)}, \theta) o_t^{(i)}(r) \quad (5.3.5)$$

As beforehand, \mathcal{W} is the hypothesis space, the i -th dimension of the t -th frame of the r -th training utterance is denoted by $o_t^{(i)}(r)$ and $K(r, w_1^N | \theta)$ is described by Equation 5.1.4.

By comparing Equations 5.3.3, 5.3.4 and 5.3.5 with Equation 5.1.2 it is clear that the statistics necessary for MBRLR transform estimation, namely the quantities $\theta_m^{\text{MBR}(i)}$ and γ_m^{MBR} , coincide with those necessary for the EBW mean update. Therefore the lattice-based MBR implementation described in Section 5.2.1 is reused to estimate these statistics for each mixture component m . Additionally, the same policy for determining the learning rate D_m in the case of the EBW updates (Section 5.2.3) is adopted for the purpose of MBRLR transform estimation. Since this is a successful approach to MBRLR transform estimation (Wang and Woodland (2004)), it is the approach used in the experimental work of Chapter 8. Since this experimental work also uses I-smoothing and complexity control, these methods are now explained in the context of MBRLR adaptation.

5.3.1 I-smoothing for MBR linear regression

To address the issue of overfitting the adaptation data, a version of the I-smoothing regularisation technique, previously introduced Section 5.2.4 in the context of MBR acoustic model estimation, has also been applied to MBR-based acoustic model adaptation in Wang and Woodland (2008). A prior upon the transform \mathbf{W} is proposed, as described by Equation 5.3.6.

$$\log p(\mathbf{W}) = \frac{\tau}{2} \sum_{m \in \mathcal{R}(s)} \sum_{r=1}^R \sum_{t=1}^{T(r)} \gamma_m(t | \hat{w}_1^{M(r)}, \mathbf{o}_1^{T(r)}, \theta) (\mathbf{o}_t(r) - \mathbf{W}\boldsymbol{\xi}_m)^\top \mathbf{C}_m^{-1} (\mathbf{o}_t(r) - \mathbf{W}\boldsymbol{\xi}_m) + k \quad (5.3.6)$$

The quantity $\gamma_m(t | \hat{w}_1^{M(r)}, \mathbf{o}_1^{T(r)}, \theta)$ is the occupancy of component m at time t given the data $\mathbf{o}_1^{T(r)}$ and the correct transcription $\hat{w}_1^{M(r)}$ and k is a normalisation term used to ensure the prior probability distribution sums to one. Note that this prior is formulated for each regression class $\mathcal{R}(s)$ and is globally maximal at the maximum likelihood estimate of the transform \mathbf{W} .

The prior probability described by Equation 5.3.6 is used in the experiments involving I-smoothed MBRLR adaptation in Chapter 8.

5.3.2 Complexity control and MBRLR adaptation

Complexity control using regression class trees has been introduced in the case of MLLR transform estimation in Section 4.3.2. In the experimental work of Chapter 8, the MBRLR transform estimation procedure adopts the same complexity control mechanism. To this end, the overall occupancies of each mixture component are accumulated in an initial pass, prior to accumulation of the statistics required for MBRLR transform estimation. These statistics are subsequently input to the MBRLR adaptation procedure to control the amount and type of MBRLR transforms estimated. Note that this ensures a fair comparison between MLLR and MBRLR since the number of transforms are identical in each case.

5.4 Summary

This chapter has presented the theory of MBR acoustic parameter estimation and adaptation. The theoretical justifications of the extended Baum-Welch parameter update equations have been reviewed. An auxiliary function for the MBR criterion has been introduced, leading to a novel derivation of the EBW update equations and a theoretical value of the learning rate D .

The lattice-based techniques deployed to capture the statistics used in the EBW update equations have been detailed and the approximations used in the context of large vocabulary continuous speech recognition have been explained. The empirical techniques used to set the learning rate have been presented, the I-smoothing regularisation technique has been introduced, and the use of acoustic scaling and lower-order language models has been motivated. This provides a basis for the discussion of Chapters 6 and 7.

With regard to MBR linear regression, the theory of MBR-based transform estimation has been presented. It has been shown that the lattice-based techniques used for implementation of the EBW update equations can be reused to capture the necessary statistics for MBRLR transform estimation. The theory and implementation details of MBRLR adaptation, including I-smoothing and complexity control, have been presented. This provides a basis for the confidence-driven refinement to MBRLR acoustic model adaptation discussed in Chapter 8.

Chapter 6

MBR error approximation

Chapter 5 discussed two approximations deployed when using the MBR criterion to estimate the acoustic model of a large vocabulary continuous ASR system. Firstly, the hypothesis space is approximated by the set of alignments specified by a word lattice. The impact of this approximation is measured experimentally in Povey (2003) where the effect of using lattices of different densities¹ is reported. Secondly, as explained in Section 5.2.1, the error associated with each lattice path is approximated. This chapter concentrates upon this second approximation and pursues Objective 2 of the thesis, as specified in Section 1.5.

The chapter is structured as follows. Section 6.1 describes previously introduced approaches to error approximation. The limitations of one of these published techniques, an alignment-based approach, referred to as the baseline approximate error, are explained in Section 6.2. Alternative alignment-based approximations are proposed in Section 6.3. These constitute a novel contribution of this thesis. The accuracy of these new approximations is compared to the accuracy of the baseline approximate error in Section 6.4. The effect of the alternative approximations upon MBR parameter re-estimation is experimentally measured in Section 6.5. A concluding discussion and propositions for future research are found in Section 6.6.

6.1 Levenshtein distance approximation

Recall from Section 5.2.1 that calculation of the Levenshtein distance between every path in the word lattice and the reference label sequence involves an impracticably large amount of computation. This is due to the need for a dynamic programming alignment of the reference label sequence and every label sequence present in the lattice. Two types of approximations are used to avoid the need for a dynamic programming alignment of each label sequence in the lattice to the reference label sequence. The first approximation will be referred to as lattice segmentation and is explained in Section 6.1.1. The second type of approximation, explained in Section 6.1.2, will be referred to as alignment-based error approximation. The

¹Lattice density is defined as the ratio of the total number of lattice arcs to the length (in seconds) of the utterance from which the lattice is derived.

latter approximation is the focus of this chapter.

6.1.1 Lattice segmentation

Lattice segmentation techniques manipulate a lattice such that each label sequence w_1^N present in the lattice is segmented into a sequence of K subsequences $w_{s_1}^{e_1} w_{s_2}^{e_2} \dots w_{s_K}^{e_K}$ where each subsequence $w_{s_i}^{e_i}$ is of length zero or more labels. Additionally the reference label sequence \hat{w}_1^M is segmented into a sequence of K subsequences $\hat{w}_{s_1}^{e_1} \hat{w}_{s_2}^{e_2} \dots \hat{w}_{s_K}^{e_K}$ of length zero or one labels. Zero-length placeholders are used to manage insertions and deletions present in lattice label sequences.

The subsequences in corresponding positions i in the segmentation, $w_{s_i}^{e_i}$ and $\hat{w}_{s_i}^{e_i}$, are then associated with each other and the approximate Levenshtein distance $L_{\text{approx}}(w_1^N, \hat{w}_1^M)$ between label sequences w_1^N and \hat{w}_1^M is defined as the sum of the Levenshtein distances between each corresponding subsequence pair, as expressed by Equation 6.1.1.

$$L_{\text{approx}}(w_1^N, \hat{w}_1^M) = \sum_{i=1}^K L(w_{s_i}^{e_i}, \hat{w}_{s_i}^{e_i}) \quad (6.1.1)$$

Since the reference subsequences $\hat{w}_{s_i}^{e_i}$ correspond to a maximum of one label, it is inexpensive to calculate the subsequence-level Levenshtein distances $L(w_{s_i}^{e_i}, \hat{w}_{s_i}^{e_i})$ and hence the calculation of the overall approximate Levenshtein distance is likewise computationally inexpensive.

The lattice segmentation process is illustrated in Figure 6.1. Here each label sequence in the original lattice is split into four subsequences, likewise the reference label sequence. Note that the deletion present in the sequence ‘INTEREST ON TIMES’ is represented by the zero-length placeholder ‘NULLWORD’.

This lattice manipulation technique was introduced with regard to MBR parameter estimation in Doumpiotis and Byrne (2004) and Doumpiotis and Byrne (2005). More details of the lattice segmentation algorithm, introduced with application to MBR-based speech decoding, are found in Goel et al. (2004).

6.1.2 Alignment-based error approximation

An alternative method is used to approximate the error of a label sequence in Povey (2003). This approximation is performed via an approximation of the accuracy of the sequence. The accuracy of a label sequence is defined as the number of correct labels minus the number of inserted labels in the sequence.

The accuracy approximation technique uses alignment information from the reference and hypothesis transcriptions as shown in Figure 6.2. A path in a lattice defines what will be referred to as a hypothesis alignment. The most likely alignment of the reference transcription is called the reference alignment. Using the hypothesis and reference alignments, the accuracy of the hypothesis label sequence is approximated using the following procedure.

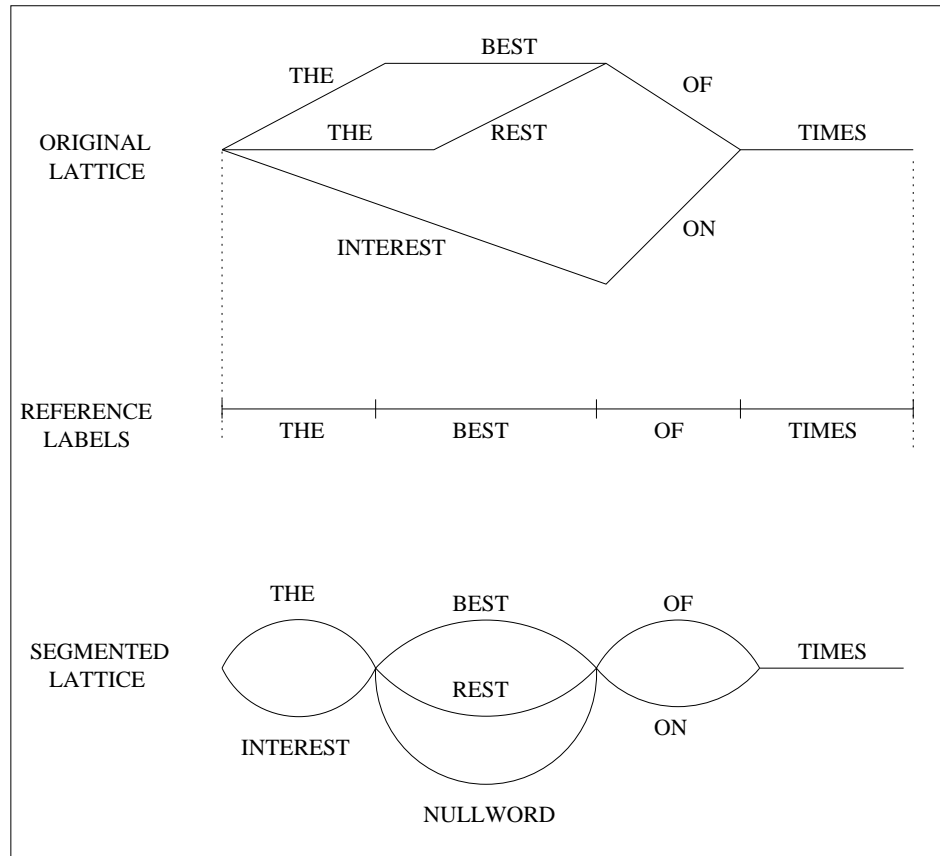


Figure 6.1: *Lattice segmentation. Each arc in the original lattice is associated with a label of the reference transcription. Null word placeholders are used to manage insertion and deletion errors. Error calculation uses the segmented lattice to avoid the need for a dynamic programming alignment of every lattice path and hence to reduce computational cost.*

Reference	A		B
Hypothesis	A		C
Length (frames)	80	20	100
Overlap proportion $e(q,z)$	0.8	0.2	1.0
$\left\{ \begin{array}{l} \text{correct: } 2e(q,z) - 1 \\ \text{incorrect: } e(q,z) - 1 \end{array} \right\}$	0.6	-0.8	0.0
$A(q)$ (max of overlapping)	0.6		0.0
Approximate accuracy		0.6	
Approximate error		1.4	
Levenshtein error		1	

Figure 6.2: *Alignment-based error approximation. Detailed in Section 6.1.2.*

For each label q in the hypothesis alignment, the set of reference labels which overlap temporally with q is identified. Then, for each overlapping reference label z , the proportion of z which overlaps with q , $e(q, z)$, is calculated. Then the accuracy of q , $A(q)$, is given by Equation 6.1.2. The overall accuracy of the hypothesis is equal to the sum of the accuracies of the comprising labels.

$$A(q) = \max_z \left\{ \begin{array}{l} -1 + 2e(q, z) \quad \text{if } q \text{ and } z \text{ are the same label} \\ -1 + e(q, z) \quad \text{if } q \text{ and } z \text{ different} \end{array} \right\} \quad (6.1.2)$$

In the case of the hypothesis in Figure 6.2 the approximated overall accuracy of the hypothesis is 0.6. The actual accuracy is 1; the number of correct labels (1) minus the number of insertions (0).

The Levenshtein distance between the reference sequence \hat{w}_1^M and hypothesis sequence w_1^N is related to the accuracy of w_1^N , $A(\hat{w}_1^M, w_1^N)$, via Equation 6.1.3, where M is the number of labels in the reference sequence. This relationship is reused to derive the approximate error $L_{\text{approx}}(\hat{w}_1^M, w_1^N)$ from the approximate accuracy, $A_{\text{approx}}(\hat{w}_1^M, w_1^N)$ say as shown in Equation 6.1.4. This alignment-based error approximation will be referred to as the baseline approximate error.

$$L(\hat{w}_1^M, w_1^N) = M - A(\hat{w}_1^M, w_1^N) \quad (6.1.3)$$

$$L_{\text{approx}}(\hat{w}_1^M, w_1^N) = M - A_{\text{approx}}(\hat{w}_1^M, w_1^N) \quad (6.1.4)$$

The baseline approximate error, like the one proposed in Doumptotis and Byrne (2004), is desirable because it circumvents the need for a dynamic programming step for each lattice

path. However this error approximation has some limitations. Section 6.2 highlights several theoretical disadvantages of the approximated error and Section 6.3 proposes alternative approaches to address some of these limitations. Note that the error approximation techniques detailed here and elsewhere in this chapter may be applied to sequences of words, phonemes or labels of other speech units.

6.2 Limitations of baseline approximate error

6.2.1 Error overestimation

The baseline approximate error, described in Section 6.1, often overestimates the error of the hypothesis label sequence. This error overestimation occurs when the label alignment times of the hypothesis and reference alignments are not identical. Figure 6.2 shows how this overestimation arises. In the example illustrated the Levenshtein error is 1, a single substitution. However the approximate error, 1.4, is larger than the actual error because the end time of the first labels of the reference and hypothesis disagree.

6.2.2 Asymmetry

Figure 6.3 shows the baseline approximate error calculation when the reference and hypothesis alignments of Figure 6.2 are swapped. The approximate error is smaller due to the correct label of the hypothesis sequence overlapping with a larger portion of the reference label in this case. This example illustrates an asymmetry in the error approximation. This asymmetry is not present in the Levenshtein error metric, and is therefore difficult to motivate in an approximation to the Levenshtein metric. Since the Levenshtein error is symmetric, it is preferable for an approximation to retain this property.

6.2.3 Insertion to deletion bias

The asymmetry of the baseline approximate error is the source of an undesirable bias with regard to the approximation of insertion and deletion errors. Figure 6.4 shows the approximate error when an insertion error occurs while Figure 6.5 displays the approximate error when the scenario is reversed and a deletion error occurs. In both cases the Levenshtein error is 1, a single insertion or deletion. Note however that the approximate error is larger in the case of the insertion error. This example illustrates a bias in the error approximation resulting in a larger error being assigned to the sequence with an insertion.

6.3 Alternative error approximations

The asymmetries discussed in Section 6.2 can be addressed by using an error approximation called the frame error. The frame error between two aligned label sequences is the number of frames at which the labels in the alignments differ, as shown in Figure 6.6. It is clear

Reference	A		C
Hypothesis	A		B
Length (frames)	80	20	100
Overlap proportion $e(q,z)$	1.0	0.17	0.83
$\left\{ \begin{array}{l} \text{correct: } 2e(q,z) - 1 \\ \text{incorrect: } e(q,z) - 1 \end{array} \right\}$	1.0	-0.83	-0.17
$A(q)$ (max of overlapping)	1.0		-0.17
Approximate accuracy		0.83	
Approximate error		1.17	
Levenshtein error		1	

Figure 6.3: *Asymmetry of alignment-based error approximation. The reference and hypothesis label sequences of Figure 6.2 are swapped, resulting in a different approximate error.*

that this measure is symmetric since the frame error between two alignments is identical regardless of which alignment is the reference.

One limitation of the frame error approximation is that an incorrect label which spans a long period of time contributes more to the overall error than an incorrect label which spans a shorter period. A second limitation is illustrated in Figure 6.7, which shows a hypothesis containing an insertion error. However, since the inserted label agrees with the reference label in the region of overlap, no error is incurred when using the frame error metric and the error and the insertion error is overlooked. The first of these limitations can be addressed via a normalisation scheme. The following section describes several such schemes.

6.3.1 Frame error normalisation

Figures 6.8 and 6.9 illustrate some different approaches to the normalisation of the frame error. Firstly the temporal region of each label of the hypothesis alignment is divided into segments corresponding to regions of overlap with different labels of the reference alignment. For example, in Figure 6.9 the label C is split into two segments, the first being associated with the label A of the reference alignment and the second associated with label B. The segment boundaries are illustrated by vertical dashed lines in Figures 6.8 and 6.9. Each segment has a corresponding frame error; the number of frames within the segment at which the hypothesis label differs from the label specified by the reference alignment. Then for each segment a normalisation factor is defined. The frame error for each segment is divided by this factor to yield a normalised frame error for each segment. The overall approximate error for the hypothesis is the sum of the normalised frame error over all segments.

Reference	A		B	
Hypothesis	A	C		B
Length (frames)	80	20	20	80
Overlap proportion $e(q,z)$	0.8	0.2	0.2	0.8
$\left\{ \begin{array}{l} \text{correct: } 2e(q,z) - 1 \\ \text{incorrect: } e(q,z) - 1 \end{array} \right\}$	0.6	-0.8	-0.8	0.6
$A(q)$ (max of overlapping)	0.6	-0.8		0.6
Approximate accuracy		0.4		
Approximate error		1.6		
Levenshtein error		1		

Figure 6.4: *Approximate error in case of insertion. Compare with Figure 6.5 which shows the approximate error in the case of a deletion.*

The reference normalised frame error (RNFE) of Figures 6.8 and 6.9 is the result of defining the normalisation factor for a segment as the length (in frames) of the overlapping reference label. In the example of Figure 6.8, a deletion error, the normalisation factor is 40 for each of the segments corresponding to the label C of the reference transcription, the length of the reference label corresponding to these segments. This leads to an accurate approximation to the Levenshtein error of 1 in this case. However in the example of Figure 6.9, an insertion error, this normalisation method yields an underestimate (0.4) of the Levenshtein error. This is because the segments corresponding to the label C of the hypothesis transcription are normalised by 100, the length of the reference label corresponding to these segments.

The hypothesis normalised frame error (HNFE) defines the normalisation factor as the length of the overlapping hypothesis label. This method leads to accurate approximation of the insertion error of Figure 6.9 but an underestimate of the deletion error of Figure 6.8.

The third normalisation technique is to normalise the frame error of each segment by the length of the shorter of the overlapping labels. This method leads to an error approximation with the desirable symmetric property and yields accurate approximations for the deletion and insertion errors illustrated in Figures 6.8 and 6.9. This approximation is referred to as the symmetrically normalised frame error (SNFE).

These approximations to the Levenshtein error are expressed by Equation 6.3.1. The approximate error between hypothesis label sequence w_1^N and reference label sequence \hat{w}_1^M is denoted by $L_{\text{approx}}(w_1^N, \hat{w}_1^M)$, $\hat{\mathcal{A}}$ represents the set of aligned reference labels corresponding to sequence \hat{w}_1^M , \mathcal{A} is the set of aligned hypothesis labels corresponding to sequence w_1^N ,

Reference	A	C	B	
Hypothesis	A	B		
Length (frames)	80	20	20	80
Overlap proportion $e(q,z)$	1.0	0.5	0.5	1.0
$\left\{ \begin{array}{l} \text{correct: } 2e(q,z) - 1 \\ \text{incorrect: } e(q,z) - 1 \end{array} \right\}$	1.0	-0.5	-0.5	1.0
$A(q)$ (max of overlapping)	1.0			1.0
Approximate accuracy		2.0		
Approximate error		1.0		
Levenshtein error		1		

Figure 6.5: *Approximate error in case of deletion. The reference and hypothesis of Figure 6.4 have been swapped. A smaller approximate error is yielded in this case.*

and $l(a, \hat{a})$ is the normalised frame error between the aligned labels a and \hat{a} .

$$L_{\text{approx}}(w_1^N, \hat{w}_1^M) = \sum_{\hat{a} \in \hat{\mathcal{A}}} \sum_{a \in \mathcal{A}} l(a, \hat{a}) \quad (6.3.1)$$

The normalised frame error between two aligned labels $l(a, \hat{a})$ is defined by Equation 6.3.2.

$$l(a, \hat{a}) = \frac{e(a, \hat{a})}{n(a, \hat{a})} \quad (6.3.2)$$

The quantity $e(a, \hat{a})$ is the number of frames at which the aligned labels a and \hat{a} differ (defined as zero if no temporal overlap exists between the aligned labels). The divisor $n(a, \hat{a})$ is the normalisation term, defined as:

- the length of \hat{a} in the case of the reference normalised frame error.
- the length of a in the case of the hypothesis normalised frame error.
- the length of the shorter of a and \hat{a} in the case of the symmetrically normalised frame error.

6.3.2 Using multiple reference alignments

The SNFE described in Section 6.3.1 addresses the asymmetry of the baseline approximate error. However the error overestimation limitation, illustrated in Figure 6.2, applies also

Reference	A		B	
Hypothesis	A	C		B
Length (frames)	80	20	20	80
Frame error	0	20	20	0
Overall frame error	40			

Figure 6.6: *Frame error metric. The number of frames at which the aligned reference and hypothesis labels differ.*

Reference	A		B	
Hypothesis	A	A		B
Overall frame error	0.0			

Figure 6.7: *Frame error metric fails to capture insertion error.*

to the SNFE. This overestimation is a consequence of differing alignment times in the hypothesis and reference label sequences. This effect may be limited by using not only one, but multiple alignments of the reference label sequence and minimising the approximate error over this set of reference alignments. This set of multiple alignments is generated using a recogniser constrained to align only the reference word sequence.

Figure 6.10 illustrates this calculation. Two reference alignments and a single hypothesis alignment are shown. The SNFE of the hypothesis sequence is calculated with respect to each of the reference alignments to give a set of errors. The minimal symmetrically normalised frame error (MSNFE) is the minimum value in this set. In the example of Figure 6.10 the SNFE with respect to Reference1 is 1.4 and the SNFE with respect to Reference2 is 1.5. Therefore the MSNFE is 1.4.

In practice a large amount of reference alignments are encoded by a reference lattice. It is therefore impractical to enumerate this set of alignments and explicitly calculate the hypothesis error with respect to each one. So a further approximation is made. This approximation minimises the error of each hypothesis label over the set of all reference alignments (instead of minimising the error of the entire hypothesis label sequence over all reference alignments). This approximation avoids the need to explicitly enumerate all reference alignments. The last row of Figure 6.10 illustrates how this approximation is applied using the two reference alignments. For the initial hypothesis label A, the upper reference

Reference	A		C	B		
Hypothesis	A			B		
Length (frames)	80	20	20	80		Overall error
Frame error	0	20	20	0		40
Reference normalised frame error	0	0.5	0.5	0		1.0
Hypothesis normalised frame error	0	0.2	0.2	0		0.4
Symmetrically normalised frame error	0	0.5	0.5	0		1.0

Figure 6.8: *Frame error normalisation in case of deletion error. In this case the reference and symmetrically normalised frame error approximations yield accurate estimates of the Levenshtein error of 1. The hypothesis normalised frame error underestimates the Levenshtein error.*

Reference	A			B		
Hypothesis	A		C	B		
Length (frames)	80	20	20	80		Overall error
Frame error	0	20	20	0		40
Reference normalised frame error	0	0.2	0.2	0		0.4
Hypothesis normalised frame error	0	0.5	0.5	0		1.0
Symmetrically normalised frame error	0	0.5	0.5	0		1.0

Figure 6.9: *Frame error normalisation in case of insertion error. In this case the hypothesis and symmetrically normalised frame error approximations yield accurate estimates of the Levenshtein error of 1. The reference normalised frame error underestimates the Levenshtein error.*

Reference1	A		B		C	Overall Error
Reference2	A	B		C		
Hypothesis	A		D	C		
Length (frames)	50	50	80	20	30	
Symmetrically normalised frame error (1)	0		1.0	0.4	0	1.4
Symmetrically normalised frame error (2)	0	0.5	1.0	0		1.5
Minimum symmetrically normalised frame error						1.4
Approx. min. symmetrically normalised frame error	0		1.0	0		1.0

Figure 6.10: *Minimal symmetrically normalised frame error (MSNFE) and approximate minimal symmetrically normalised frame error (AMSNFE). MSNFE minimises the SNFE of the entire hypothesis sequence over the set of reference alignments. AMSNFE minimises the SNFE of each hypothesis label individually over the set of reference alignments.*

alignment Reference1 provides the minimal error. In the case of the second hypothesis label D, both reference alignments induce the same error. The error for the third hypothesis label C is minimised using the lower reference alignment Reference2. Summing the minimal error over all hypothesis labels gives the approximate minimal symmetrically normalised frame error (AMSNFE). Note that in the example illustrated in Figure 6.10 the AMSNFE of 1.0 is a closer approximation to the Levenshtein error (1) of the hypothesis sequence than the MSNFE of 1.4. This is a consequence of the flexibility of the former technique to select different reference alignments to minimise the error of different labels of the same hypothesis.

6.4 Error approximation analysis

The theoretical arguments presented in Sections 6.2 and 6.3 are experimentally tested in this section. Approximately one hour of speech comprising 1851 utterances is selected from a training corpus of spontaneous meeting speech. This corpus is described in Section B.1.5. Each utterance is recognised using the first and second pass of the recognition system described in Section B.1 to produce a phoneme-level hypothesis alignment for each utterance. The reference phoneme sequence for each utterance is aligned using the same recognition system. To measure the AMSNFE approximation, a lattice of reference alignments is also generated by the recognition system at this stage.

For each utterance, the phoneme-level Levenshtein error is calculated using dynamic programming alignment, and the phoneme-level approximate error is calculated using each

of the techniques described in Sections 6.2 and 6.3.

6.4.1 Raw error approximation

From the dataset described above, the utterances which are transcribed with non-zero phoneme-level error are selected. This subset is then further split into three smaller subsets, one which contains those utterances transcribed with substitution errors only (\mathcal{S}), one which contains those utterances transcribed with insertion errors only (\mathcal{I}) and one which contains those utterances with deletion errors only (\mathcal{D}).

Table 6.1 presents some analysis of the approximate error techniques described in this chapter. Each row of Table 6.1 corresponds to the subset of utterances indicated in the first column, where the notation of the previous paragraph has been used. The number of utterances in each subset is displayed in the second column and the sum of the Levenshtein errors of each transcribed utterance in each set is indicated in the third column. The remaining columns display the sum of the approximate errors of each transcribed utterance in each set for each approximation technique. The acronyms BAE and FE are used to represent the baseline approximate error and the frame error, respectively.

Set	# Utt	Lev	Approximate error					
			BAE	FE	HNFE	RNFE	SNFE	AMSNFE
\mathcal{S}	110	245	560.9	3211	345.4	359.1	426.4	290.8
\mathcal{I}	43	82	208.2	1317	115.8	70.9	128.7	80.3
\mathcal{D}	184	327	552.0	2297	193.9	340.5	370.2	181.2

Table 6.1: *Analysis of error approximations for substitution, insertion and deletion errors.*

Discussion

The results presented in Table 6.1 are used to test the theoretical arguments presented in this chapter. Firstly, comparing the numbers in third and fourth columns, it is clear that the baseline approximate errors are greater than the true Levenshtein errors for each utterance subset. This confirms that the BAE approximation overestimates errors, as discussed in Section 6.2.1.

Note further that the ratio of the BAE to the Levenshtein error is 2.54 (208.2/82) in the case of insertion errors, and 1.68 (552.0/327) in the case of deletion errors. The higher ratio in the case of insertions is evidence of the insertion to deletion bias discussed in Section 6.2.3.

Examination of the same ratios in the case of the FE approximation reveals that the FE also possesses an insertion to deletion bias. The ratio is larger (16.1 (1317/82)) in the case of insertion errors than in the case of deletions (7.02 (2297/327)). This somewhat surprising result is due to the fact that, on average, for the dataset considered, deletion errors correspond to phonemes of shorter duration than phonemes associated with insertion errors.

In concurrence with the arguments of Section 6.3.1, it is clear from Table 6.1 that the RNFE approximation underestimates insertion errors (an approximation of 70.9, in comparison with a true insertion error of 82), while the HNFE approximation underestimates deletion errors (an approximation of 193.9, in comparison with a true deletion error of 327).

Again, in accordance with the discussion of Section 6.3.1, the results of Table 6.1 show that the SNFE does not underestimate either deletion or insertion errors. The ratio of the SNFE approximation to true Levenshtein error is 1.57 ($128.7/82$) in the case of insertions and 1.13 ($370.2/327$) in the case of deletions, showing that some insertion to deletion bias is still present even when using the SNFE. This effect is due to the fact that insertion errors sometimes correspond to phonemes of longer duration than the overlapping phonemes of the reference transcription. Such insertion errors are then normalised by the length of a relatively short reference phoneme, resulting in an overestimation of the insertion error. Although the analogous phenomenon occurs also in the case of deletion errors (in this case, normalisation by the length of a relatively short hypothesis phoneme occurs), it happens less often than in the case of insertions. This in turn is because, on average, for the dataset considered, deletion errors correspond to phonemes of shorter duration than phonemes associated with insertion errors.

Comparing the entries in the last and second last columns of Table 6.1, it can be seen that the AMSNFE approximation always yields a lower value than that of the SNFE approximation. However, a large underestimation of deletion errors occurs in this case. Examination of a typical set of reference alignments provides an explanation of this phenomenon. The set of reference alignments often contains two alternative alignments for a word, one of which contains some word-end silence and the other which contains little or none, as shown in Figure 6.11. For the dataset considered, the most likely (i.e. the 1-best) alignment is usually the reference which contains silence. When using the SNFE approximation, the most likely alignment is used as the reference.

Reference1 (more likely)	h	ay	sil
Reference2	h	ay	
Hypothesis	h	ay	
Length (frames)	80	50	30
Symmetrically normalised frame error (1)	0	0	1
Symmetrically normalised frame error (2)	0	0	0
Approx. min. symmetrically normalised frame error	0	0	0

Figure 6.11: *Silence handling using the approximate minimal symmetrically normalised frame error.*

In the example of Figure 6.11, the SNFE of the hypothesis is 1 due to the deletion of a silence label. When using multiple reference alignments, as in the case of the AMSNFE approximation, a hypothesis which erroneously deletes word-end silence may be assigned zero error due to the presence of the additional, less likely, reference alignment. In the example of Figure 6.11, the AMSNFE approximation assigns zero error to the hypothesis due to the presence of the reference alignment Reference2. This accounts for the underestimation of deletion errors witnessed in the case of the AMSNFE approximation.

6.4.2 Error approximation accuracy

The accuracy of the error approximations described in Sections 6.2 and 6.3 is now measured. Note that there exist several ways to quantify this accuracy, for example the summed squared difference between the approximate errors and the true errors. In this section, the correlation between the true Levenshtein error and the approximate error is used as a measure of the accuracy of the approximation. This correlation is used because it is a measure of the similarity between the MBR criterion which uses the error approximation (the approximate MBR criterion, say) and the MBR criterion which uses the true Levenshtein error (the true MBR criterion, say). An approximation which correlates perfectly with the Levenshtein error yields an approximate MBR criterion which, in turn, yields the same model parameter updates as the true MBR criterion.

To measure the correlation of the approximate error with the true Levenshtein error, the entire 1851-utterance subset of the training corpus described above is used. The correlation of the phoneme-level Levenshtein and approximate errors is then calculated for each of the error approximations.

Figure 6.12 plots the approximate error against the Levenshtein error in the cases of the baseline approximate error and the SNFE approximation. It is noticeable from these plots that the SNFE yields a higher correlation with the Levenshtein error than that yielded by the baseline approximate error. Table 6.2 records the correlation of the different error approximations with the Levenshtein error.

Approximation method	Correlation with Levenshtein
Frame error	0.909
Baseline approximate error	0.959
Reference normalised frame error	0.968
Hypothesis normalised frame error	0.938
Symmetrically normalised frame error	0.976
Approx. min. symm. normalised frame error	0.986

Table 6.2: *Correlation of error approximations with Levenshtein error.*

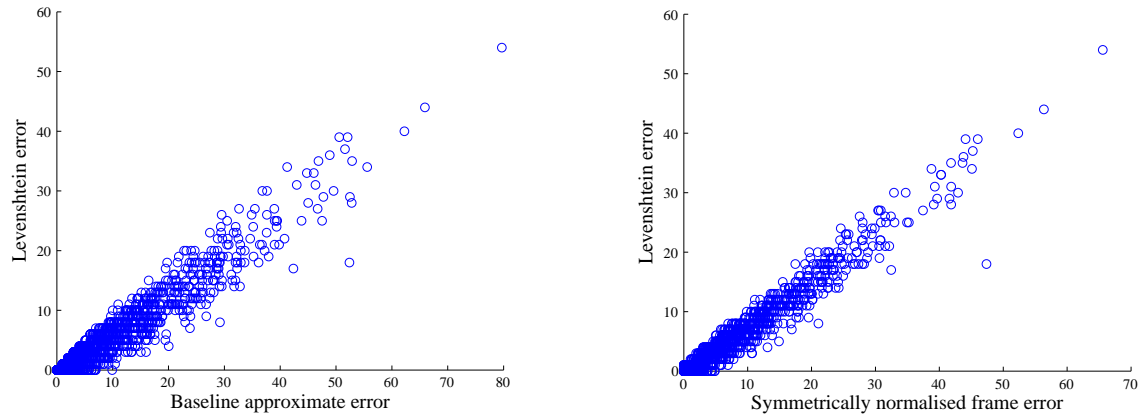


Figure 6.12: *Correlation of error approximations with Levenshtein error. Each point represents an utterance. The approximate error is plotted against the Levenshtein error for each utterance in the cases of the baseline approximate error and the symmetrically normalised frame error. The correlation coefficient between the baseline approximate error and the Levenshtein error is 0.959. The correlation coefficient between the symmetrically normalised frame error and the Levenshtein error is 0.976.*

Discussion

It is firstly worth noting that the differences between all pairs of correlation coefficients displayed in Table 6.2 are all significant. A significant difference between two correlation coefficients is indicated by the procedure described in Steiger (1980) for correlation coefficients derived from the same sample. This procedure assumes that the samples are independently selected, a reasonable assumption for the dataset considered here.

Table 6.2 shows that the frame error displays weaker correlation with the Levenshtein error than all other approximations. This observation indicates that the unnormalised frame error is a relatively inaccurate approximation of the Levenshtein error.

The RNFE has a higher correlation than the HNFE approximation. This result is not immediately understandable, but is due to the nature of the ASR system used to provide hypotheses in this evaluation. The ASR system is designed to output more deletion errors than insertion errors. As argued in Section 6.3.1, the RNFE captures deletion errors more effectively than the HNFE approximation. Consequently, the RNFE is a better approximation of the Levenshtein error for this particular dataset.

The SNFE approximation displays a higher correlation with the Levenshtein error than both the hypothesis and reference normalised frame error. So, with respect to this measurement of accuracy, the SNFE is a more accurate approximation of the Levenshtein error than both the RNFE and HNFE.

Despite its underestimation of deletion errors, as discussed in Section 6.4.1, the AM-SNFE metric yields a correlation coefficient of 0.986, greater than that of the SNFE metric.

This result suggests that incorporation of knowledge of multiple reference alignments improves the SNFE approximation of the Levenshtein distance.

6.5 Evaluation: MBR-estimated acoustic models

The impact of each of the error approximations described in Section 6.3 upon the behaviour of MBR-estimated acoustic models is now measured. The phoneme-level MBR criterion (see Section 3.2.5) is used in order to give a comparison with the standard MPE criterion.

The task used in this evaluation is the large vocabulary transcription of meeting speech. Appendix B.2 describes the recognition system used to evaluate the MBR-estimated acoustic models. Appendix B.3 describes the MBR parameter estimation procedure. The training dataset used is the 104 hour training dataset described in Section B.1.5. To evaluate the effect of the different error approximations, the phoneme-level MBR-estimated acoustic models are substituted into the second recognition pass of the transcription system. The *rt05eval* NIST evaluation dataset (see Section B.1.5 for details) is used as test data.

6.5.1 Unsmoothed MBR

Figure 6.13 displays how the recognition WER yielded by the phoneme-level MBR-estimated models varies with each iteration of parameter estimation. No smoothing of the MBR criterion is performed. As shown in the legend of Figure 6.13, each plotted curve corresponds to the use of a different error approximation within the MBR implementation. The zeroth iteration corresponds to the performance of the baseline ML-estimated models.

All of the error approximations used result in MBR re-estimated models which display improved classification performance over the ML-estimated models. The models which generalise best are generally yielded after three iterations of MBR re-estimation after which the effects of overfitting the training data become evident. Table 6.3 provides some analysis of the errors made by the models yielded after three MBR iterations.

Discussion

The initial row of Table 6.3 displays the error analysis for the ML-estimated models used as the starting point for MBR estimation. The remaining rows analyse the errors committed by MBR-estimated models. Note firstly that, with the exception of the RNFE approximation method, all MBR-estimated models display deletion rates greater than that of the ML-estimated models. This is because, as discussed in Section 6.4.1, for the dataset considered, an insertion to deletion bias is manifested when using all error approximations other than the RNFE approximation.

The analysis of Table 6.3 shows that the RNFE approximation technique yields the least number of deletion errors but the greatest number of insertion errors of all the error approximations used. This is due to the underestimation of insertion errors when using the RNFE technique, as explained in Section 6.3.1.

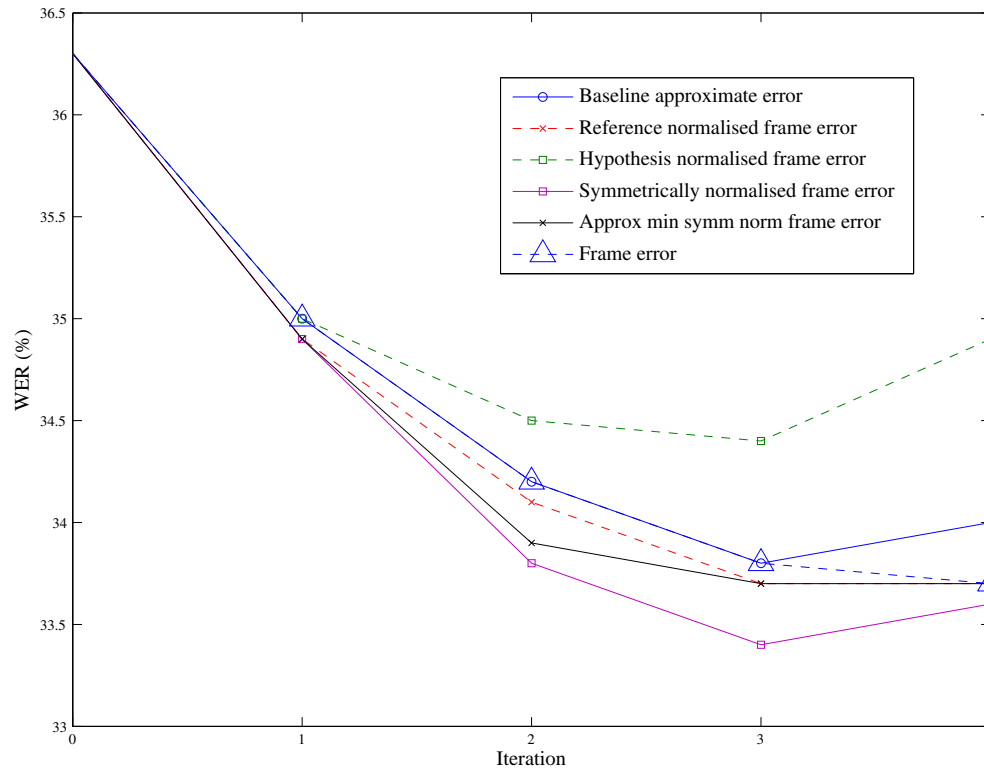


Figure 6.13: Performance of unsmoothed MBR-estimated models using different error approximations (rt05seval dataset). The error approximation used is indicated in the legend.

The HNFE method gives rise to acoustic models which yield the highest deletion rate and lowest insertion rate of all the approximations considered. This is due to the systematic underestimation of deletion errors when using this technique, again explained in Section 6.3.1.

The BAE technique yields models which have a relatively high deletion rate and low insertion rate. This is explained by the insertion to deletion bias discussed in Section 6.2.3.

The use of the SNFE approximation is inherently unbiased with regard to the correction of insertion and deletion errors. Moreover, this technique yields WER improvements over all other approximation methods including the baseline error approximation. A statistical test reveals that these improvements are significant. Note that here, and elsewhere in this thesis, a significant improvement is defined as significant at the 95% confidence level using the matched pairs sentence segment word error test (MPSSWE) (Gillick and Cox (1989),

Criterion	Approximation method	Sub (%)	Del (%)	Ins (%)	WER (%)
ML	-	18.1	13.6	4.7	36.4
MBR	FE	15.9	14.3	3.7	33.8
	BAE	15.7	14.6	3.5	33.8
	RNFE	16.2	13.5	4.1	33.7
	HNFE	16.0	15.2	3.2	34.4
	SNFE	15.6	14.2	3.6	33.4
	AMSNFE	15.7	14.4	3.6	33.7

Table 6.3: *Performance analysis of models yielded after three iterations of unsmoothed MBR estimation (rt05seval dataset).*

Pallett et al. (1990)).

Despite having displayed a higher correlation with the Levenshtein error (see Table 6.2), the AMSNFE technique is significantly out-performed in terms of WER by the SNFE method. This is due to the underestimation of deletion errors when using the AMSNFE technique, as discussed in Section 6.4.1. This is reflected in the comparatively high deletion rate of the AMSNFE technique, as displayed in Table 6.3.

6.5.2 I-smoothed MBR

To investigate if the classification performance improvements yielded by the SNFE approximation over the baseline approximate error persist when using smoothed MBR criteria, the experiment described above is repeated using I-smoothed phoneme-level MBR (see Section 5.2.4). An I-smoothing factor (τ^I) equal to 50 is used.

Approximation Method	Sub (%)	Del (%)	Ins (%)	WER (%)
Baseline approximate error	15.8	14.6	3.3	33.7
Symmetrically normalised frame error	15.7	14.3	3.4	33.4

Table 6.4: *Performance analysis of models yielded after five iterations of I-smoothed MBR estimation (rt05seval dataset). An I-smoothing factor τ^I of 50 is used.*

Figure 6.14 displays how the recognition WER yielded by the smoothed phoneme-level MBR-estimated models varies with each iteration of parameter estimation in the cases of the baseline approximate error and SNFE approximations. Again, the zeroth iteration corresponds to the performance of the baseline ML-estimated models. Table 6.4 analyses the errors made by the models yielded after five smoothed MBR iterations in the case of each error approximation technique.

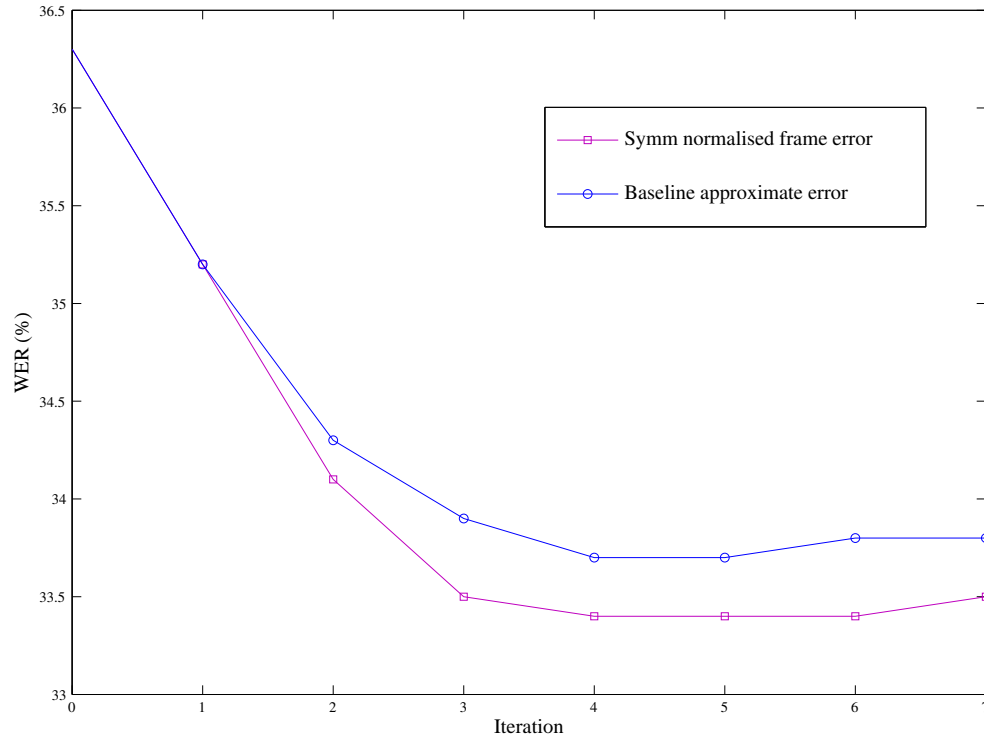


Figure 6.14: Performance of I -smoothed MBR-estimated models using different error approximations (rt05seval dataset). The error approximation used is indicated in the legend and the I -smoothing factor τ^I is set to 50.

Discussion

Although the effects of overfitting the training data are alleviated in later iterations (compare the performance of the fourth iteration models in the smoothed and unsmoothed cases), the performance of the fifth iteration models in the smoothed case is similar to the performance of the third iteration models in the unsmoothed case.

Examination of Table 6.4 reveals that, as in the case of unsmoothed MBR model estimation, the baseline approximate error technique yields models which have a relatively high deletion rate and relatively low insertion rate. As in the case of unsmoothed MBR model estimation, the SNFE approximation yields models which in turn yield a lower WER than those models estimated using the baseline error approximation. The MPSSWE test reveals that the WER improvements yielded by models estimated with the SNFE approximation are significant.

6.6 Summary and future work

Limitations of a previously introduced alignment-based error approximation technique have been highlighted in this chapter. Alternative approximations based on the frame error metric have been introduced. The accuracy of these approximations has been evaluated, as well as their impact upon the performance of MBR acoustic model re-estimation. Significant improvements over the previously introduced error approximation have been recorded for a large vocabulary recognition task when the symmetrically normalised frame error approximation is deployed for MBR acoustic parameter re-estimation.

6.6.1 Future work

While the use of multiple reference alignments has shown some promise with regard to accurate approximation of the Levenshtein error, the technique has failed to deliver performance improvements when deployed for MBR acoustic parameter estimation. Future work may consider more careful usage of multiple reference alignments to gain an error approximation which does not underestimate deletion errors.

While this chapter has compared the impact of different alignment-based error approximations, there has been no comparison between these techniques and the lattice segmentation methods discussed in Section 6.1. It is not clear how these very different approaches to error approximation compare and future research should analyse the differences between these methods.

Chapter 7

Sub-word MBR criteria

7.1 Introduction

Much of the previous research into the use of the MBR criterion for acoustic model parameter estimation in speech recognition (e.g. Kaiser et al. (2002), Doumpiotis and Byrne (2004)) has defined the criterion using the set of word sequences as the hypothesis space. As mentioned in Section 3.2.5, an alternative formulation of the MBR criterion is introduced in Povey (2003). This formulation, called the minimum phone error (MPE) criterion, uses the set of competing phoneme sequences as the hypothesis space and a corresponding phoneme-level error function. Experiments reported in Povey (2003) show that use of the phoneme-level formulation yields acoustic models which display small improvements over the test set performance of word-level MBR-estimated acoustic models on the Switchboard large vocabulary conversational speech transcription task. However the following questions remain unanswered.

- A. What is the motivation behind use of a phoneme-level MBR criterion instead of word-level MBR?
- B. Is the improved generalisation of phoneme-level MBR-estimated acoustic models over word-level MBR-estimated models due to different treatment of homophones and heteronyms¹ of reference words?
- C. Is the improved generalisation of phoneme-level MBR-estimated acoustic models over word-level MBR-estimated models due to the incorporation of errors related to word-end silence in the case of the phoneme-level criterion?
- D. Neither word-level nor phoneme-level MBR exploit information about errors at the lower levels of acoustic modelling. Are other sub-word-level MBR criteria motivated?

¹For the purposes of the arguments in this chapter, homophones are defined as words which are spelled differently but have identical phonemic representation, e.g. ‘see’ and ‘sea’. Heteronyms are words which share the same spelling but which are pronounced differently.

- E. Do significant differences exist between the test set performance of acoustic models estimated using the word and sub-word-level criteria?

The purpose of this chapter is to pursue thesis Objective 3 (see Section 1.5) by addressing these questions. The chapter is structured as follows. In response to question A, the use of the phoneme-level MBR formulation is motivated in Section 7.2. The differences between the phoneme-level MBR criterion and the word-level MBR criterion, in terms of the treatment of homophones, heteronyms and word-end silence are explained.

With regard to question B, a word-level MBR criterion which treats homophones and heteronyms of reference words in the same way as the phoneme-level criterion, referred to as the phoneme-sensitive word-level MBR criterion, is introduced in Section 7.3. The generalisation of the phoneme-sensitive word-level MBR criterion is later compared to that of phoneme and word-level MBR criteria in Section 7.7.

To address question C, a word-level MBR criterion which incorporates errors related to word-end silence, referred to as the silence-sensitive word-level MBR criterion, is introduced in Section 7.4. The generalisation of the silence-sensitive word-level MBR criterion is later compared to that of phoneme and word-level MBR criteria in Section 7.7.

With respect to Question D, novel acoustic model-level MBR criteria are introduced and motivated in Section 7.5. Finally, to answer question E, the test set performance of acoustic models estimated using the different MBR formulations introduced in this chapter are compared in Section 7.7. Analysis of the similarity of the MBR criteria helps to interpret the results of Section 7.7 and is presented beforehand in Section 7.6. A concluding discussion and a proposed direction for future research are found in Section 7.8.

7.2 Motivating phoneme-level MBR

The phoneme-level MBR criterion may be motivated in two different ways, discussed separately in Sections 7.2.1 and 7.2.2.

7.2.1 Focus on acoustic confusion

Discriminative model estimation attempts to adjust model parameters to differentiate the correct category from the set of competing categories. In the context of ASR, a desirable property of a (discriminative) model estimation technique is to increase the posterior probability of the correct model sequence and to decrease the posterior of competing sequences.

With this in mind, consider the example shown in Figure 7.1. The reference word sequence has posterior probability of 0.3, as does the competing hypothesis Hyp1. A second hypothesis, Hyp2, has posterior probability of 0.4. Suppose that the acoustic models comprising the word sequence ‘NOT TOO FAIR’ are identical to those comprising ‘KNOT TWO FARE’. This is the case in a typical large vocabulary ASR system. Suppose further that these word sequences have identical language model probabilities. Then these two word sequences have identical posterior probabilities, regardless of the acoustic model parameters. Recall that MBR acoustic model estimation adjusts the acoustic model parameters

			Levenshtein error	Posterior	Updated posterior
WORD-LEVEL	Reference	NOT TOO FAIR	0	0.3	0.0
	Hyp1	KNOT TWO FARE	3	0.3	0.0
	Hyp2	NOT TOO CARE	1	0.4	1.0
PHONEME-LEVEL	Reference	n aa t t uw f ey r	0	0.3	0.5
	Hyp1	n aa t t uw f ey r	0	0.3	0.5
	Hyp2	n aa t t uw k ey r	1	0.4	0.0

Figure 7.1: Comparison of phoneme and word-level MBR criteria in the case of overlapping homophones. The acoustic models comprising the word sequence ‘NOT TOO FAIR’ are identical to those comprising ‘KNOT TWO FARE’. Under the additional assumption that the language model probabilities of these sequences are identical, the sequences always have identical posterior probability. Suppose that the posterior is as shown in the column labelled ‘Posterior’. The word-level MBR criterion achieves its minimum by adjusting the posterior probabilities as shown in the column labelled ‘Updated posterior’. Note that the posterior probability of the reference transcription is decreased to 0. Compare this with the effect of the phoneme-level MBR criterion. Since the word sequences ‘NOT TOO FAIR’ and ‘KNOT TWO FARE’ are phonemically identical, the phoneme-level MBR criterion achieves its minimum by reducing the posterior probability of the second hypothesis to 0. Note that the posterior probability of the reference transcription is increased to 0.5.

such that the criterion is minimised. So, given that the posterior of Hyp1 and the reference word sequence are equal, the word-level MBR criterion achieves its minimum by altering the acoustic model to produce the posterior probabilities as shown in the column labelled ‘Updated posterior’ in Figure 7.1. Notice that the posterior probability of the reference model is reduced. As discussed above, this is an undesirable property of a discriminative training criterion. This example therefore highlights a theoretical problem with the word-level MBR criterion.

Compare the behaviour of phoneme-level MBR using the same example of Figure 7.1. Notice that the reference word sequence is phonemically identical to the hypothesis Hyp1. In the case of model re-estimation to minimise the phoneme-level MBR criterion, the posterior probabilities are adjusted as shown in the column labelled ‘Updated posterior’. The posterior probability of the hypothesis Hyp2 is reduced while the posterior of hypothesis Hyp1 and, importantly, the reference sequence, is increased.

This example illustrates the following difference between the word and phoneme-level MBR criteria. The phoneme-level MBR criterion focusses on the resolution of acoustic confusions and makes no attempt to resolve confusions which are beyond the remit of

the acoustic model. In the example of Figure 7.1, discrimination between the reference transcript and Hyp1 can only be resolved via the language model since the acoustic models of these sequences are identical. Focus on acoustic confusion is a desirable property of an acoustic model training criterion and provides motivation for the use of phoneme-level MBR over word-level MBR.

The example illustrated in Figure 7.1 used homophones of words in the reference transcription to illustrate behaviour differences between the word and phoneme-level MBR criteria. As will now be demonstrated, heteronyms may also be used to illustrate differences between these criteria.

			Levenshtein error	Posterior	Updated posterior
WORD-LEVEL	Reference	LEAD BOW	0	0.5	0.5
	Hyp1	LEAD BOW	0	0.5	0.5
PHONEME-LEVEL	Reference	l iy d b ow	0	0.5	1.0
	Hyp1	l eh d b aw	2	0.5	0.0

Figure 7.2: *Comparison of phoneme and word-level MBR criteria in the case of overlapping heteronyms. The reference and hypothesis phoneme-level transcription of the word sequence ‘LEAD BOW’ differ, corresponding to different pronunciations (heteronyms) of the words ‘LEAD’ and ‘BOW’. Suppose that the posterior is as shown in the column labelled ‘Posterior’. The phoneme-level MBR criterion achieves its minimum by adjusting the posterior probabilities as shown in the column labelled ‘Updated posterior’. Note that the posterior probability of the reference transcription is increased to resolve the acoustic confusion present in the illustrated scenario. The word-level MBR criterion is unaware of this confusion and delivers no update to the parameters to adjust the posterior probabilities.*

Figure 7.2 shows the reference transcription ‘LEAD BOW’ and a hypothesis of equal posterior probability which corresponds to the same spelling but different pronunciation of this transcription. The column labelled ‘Updated posterior’ shows how the word and phoneme-level MBR criteria adjust the posterior probability of the reference and hypothesis transcriptions to achieve their respective minima. Notice that the phoneme-level criterion adjusts the model parameters to resolve the acoustic confusion. In contrast, the word-level criterion is unaware of this confusion, resulting in no adjustment of the model parameters.

Analysis

To give an indication of the practical importance of the arguments presented above, the 64 hour training dataset used in the experiments of Section 7.7 is analysed. Each utterance of this dataset has an associated reference alignment and lattice used in the MBR acoustic model estimation procedure. The first row of Table 7.1 displays the total number of lattice arcs associated with the dataset, and the overall value of the word and phoneme-level MBR criteria. The second row shows how many of these arcs represent words which overlap temporally with a homophone in the reference alignment, and the contribution of these arcs (i.e. the posterior-weighted sum of errors over these arcs) to the word and phoneme-level MBR criteria. The third row of the table displays the same information for arcs which overlap temporally with a heteronym in the reference alignment. The SNFE approximation is used to approximate errors in these calculations.

	#Arcs	Criterion contribution	
		Word-level	Phoneme-level
Total	49463799	397055	892891
Homophones	1116144	21198	3663
Heteronyms	1621116	1097	14466

Table 7.1: *Relative contribution of temporally overlapping homophones and heteronyms of reference words to the word and phoneme-level MBR criteria.*

A small percentage of all lattice arcs, 2.26%, overlap temporally with a homophone of their associated word in the reference alignment. A slightly larger amount of arcs, 3.28%, overlap with a heteronym of their associated word. In the case of the word-level MBR criterion, overlapping homophones account for 5.34% (21198/397055) of the total criterion value. This is relatively high in comparison to the 0.41% (3663/892891) contribution of overlapping homophones in the case of the phoneme-level criterion. This demonstrates that, as illustrated in Figure 7.1, homophones are assigned a relatively low error in the case of the phoneme-level criterion.

The overlapping heteronyms contribute to only 0.28% (1097/397055) of the total word-level MBR criterion value, whilst accounting for 1.62% of the total phoneme-level MBR criterion value. This shows that, as illustrated in Figure 7.2, heteronyms are assigned a relatively high error in the case of the phoneme-level criterion.

7.2.2 Additional word-end silence discrimination

The large vocabulary ASR systems used in the experimental work of this thesis model each word with optional silence at the end. In the implementation of phoneme-level MBR used throughout this thesis, silence labels are treated in the same way as phoneme labels. Given this, word and phoneme-level MBR criteria differ with regard to their treatment of word-end silence. This difference is illustrated in Figure 7.3, where the word and phoneme-level alignments of a reference and hypothesis sequence are shown. The word-level error of the

hypothesis is zero. However, the phoneme-level error of the hypothesis is one, due to the deletion of the label ‘sp’ representing silence at the end of the word ‘THE’.

Reference (word)	<s>	THE				BEST				</s>
Reference (phoneme)	sil	dh	ax	sp	b	eh	s	t	sil	
Hypothesis (word)	<s>	THE				BEST				</s>
Hypothesis (phoneme)	sil	dh	ax		b	eh	s	t	sil	
Symmetrically normalised frame error (word)		0				0				0
Symmetrically normalised frame error (phoneme)		1				0				0

Figure 7.3: *Additional word-end silence discrimination using the phoneme-level MBR criterion. The word and phoneme-level alignments of a reference and hypothesis sequence are shown. Note that the symbols ‘<s>’ and ‘</s>’ represent silence at the beginning and end of an utterance, respectively. The word-level error of the hypothesis is zero. However, the phoneme-level error of the hypothesis is one, due to the deletion of the label ‘sp’ representing silence at the end of the word ‘THE’.*

In this way, the phoneme-level MBR criterion discriminates between those hypotheses which agree and those which disagree with the reference label sequence with regard to word-end silence. The word-level MBR criterion fails to discriminate in this way as it respects only word-level labels. Therefore phoneme-level MBR may be additionally motivated by noting that it naturally provides additional discrimination with regard to word-end silence. Note that some discrimination between silence and non-silence is built into the word-level MBR criterion due to the presence of words representing silence. However such words are placeholders constrained to be either at the utterance start or end, and inter-word silence is not represented at the word-level.

Analysis

To illustrate the practical relevance of the argument above, the contribution of word-end silence errors to the phoneme-level MBR criterion value is measured. The 64 hour training dataset, reference alignments, and lattices used in the experiments of Section 7.7 are used for this analysis. Table 7.2 displays the overall value of the phoneme-level MBR criterion, and the contribution of word-end silence errors to this value (i.e. the posterior-weighted sum of errors arising from the confusion of word-end silence with non-silence). A considerable

proportion (7.97%) of the phoneme-level MBR criterion is due to word-end silence errors.

Phoneme-level criterion	Word-end silence contribution
892891	71157

Table 7.2: *Contribution of word-end silence errors to the phoneme-level MBR criterion.*

7.3 Phoneme-sensitive word-level MBR criterion

As explained in Section 7.2.1, the word and phoneme-level MBR criteria differ with respect to their treatment of overlapping homophones and heteronyms of reference words. To measure the impact of this difference upon the performance of the resulting acoustic models, a word-level MBR criterion is constructed which backs off to the behaviour of the phoneme-level MBR criterion for overlapping homophones and heteronyms of reference words. This criterion assigns the phoneme-level SNFE approximate error to hypothesis words which overlap a homophone or heteronym in the reference alignment, and assigns the word-level SNFE approximate error to all other words.

Reusing the notation of Section 6.3.1, this error function is defined in Equation 7.3.1, where $l^{\text{ph}}(a, \hat{a})$ is defined by Equation 7.3.2.

$$L_{\text{approx}}(w_1^N, p_1^{N_p}, \hat{w}_1^M, \hat{p}_1^{M_p}) = \sum_{\hat{a} \in \hat{\mathcal{A}}} \sum_{a \in \mathcal{A}} l^{\text{ph}}(a, \hat{a}) \quad (7.3.1)$$

$$l^{\text{ph}}(a, \hat{a}) = \begin{cases} \alpha_p L_{\text{approx}}^{\text{nosil}}(p_a, \hat{p}_a) & \text{if } a \text{ overlaps a homophone or heteronym in the reference} \\ \alpha_w l(a, \hat{a}) & \text{otherwise} \end{cases} \quad (7.3.2)$$

In the above equations, the word-level and phoneme-level hypothesis label sequences are represented by w_1^N and $p_1^{N_p}$ respectively. The word-level and phoneme-level reference label sequences are denoted by \hat{w}_1^M and $\hat{p}_1^{M_p}$ respectively. The set $\hat{\mathcal{A}}$ is the set of word-level aligned reference labels corresponding to sequence \hat{w}_1^M , and \mathcal{A} is the set of aligned word-level hypothesis labels corresponding to sequence w_1^N . The error $l(a, \hat{a})$ is the SNFE between the aligned word-level labels a and \hat{a} , and the error $L_{\text{approx}}^{\text{nosil}}(p_a, \hat{p}_a)$ is the SNFE between those aligned phoneme-level hypothesis and reference labels which overlap the word-level label a , excluding any errors induced by labels which represent word-end silence.

The criterion corresponding to the error function of Equation 7.3.1 will be referred to as the phoneme-sensitive word-level MBR criterion. The scalars α_w and α_p are the word-level and phoneme-level weighting factors of Table 7.4, respectively. Use of these weights ensures that the influence of errors incurred by overlapping heteronyms and homophones of reference words is equal for the cases of the phoneme-level and phoneme-sensitive word-level MBR criteria.

The similarity of the phoneme-level and phoneme-sensitive word-level MBR criteria is measured in Section 7.6. The test set performance of phoneme-level and phoneme-sensitive

word-level MBR-estimated acoustic models are then compared in Section 7.7. These experiments provide an answer to question B of Section 7.1.

7.4 Silence-sensitive word-level MBR criterion

To measure the impact of the additional word-end silence discrimination present in the phoneme-level MBR criterion, a silence-sensitive word-level MBR criterion is formulated. The silence-sensitive word-level MBR criterion is designed to incorporate discrimination between silence and non-silence which is absent in the standard word-level MBR criterion and present in the phoneme-level MBR criterion. This criterion is defined as the MBR criterion which deploys the error function defined by Equation 7.4.1.

$$L_{\text{approx}}(w_1^N, p_1^{N_p}, \hat{w}_1^M, \hat{p}_1^{M_p}) = \alpha_w L_{\text{approx}}(w_1^N, \hat{w}_1^M) + \alpha_p L_{\text{approx}}^{\text{sil}}(p_1^{N_p}, \hat{p}_1^{M_p}) \quad (7.4.1)$$

In Equation 7.4.1, $L_{\text{approx}}(w_1^N, \hat{w}_1^M)$ is the SNFE approximation between the word-level alignment of word sequence w_1^N and the word-level alignment of reference word sequence \hat{w}_1^M . The phoneme sequence $p_1^{N_p}$ is the phoneme sequence of highest likelihood, given the word sequence w_1^N . The phoneme sequence $\hat{p}_1^{M_p}$ is the phoneme sequence of highest likelihood, given the reference word sequence \hat{w}_1^M . The error $L_{\text{approx}}^{\text{sil}}(p_1^{N_p}, \hat{p}_1^{M_p})$ is an adjusted SNFE between $p_1^{N_p}$ and $\hat{p}_1^{M_p}$ which respects only silence labels, as described by Equation 7.4.2.

$$L_{\text{approx}}^{\text{sil}}(p_1^{N_p}, \hat{p}_1^{M_p}) = \sum_{\hat{a} \in \hat{\mathcal{A}}} \sum_{a \in \mathcal{A}} l^{\text{sil}}(a, \hat{a}) \quad (7.4.2)$$

In Equation 7.4.2, $\hat{\mathcal{A}}$ and \mathcal{A} represent the sets of aligned phoneme-level reference and hypothesis labels corresponding to the most likely alignments of phoneme sequences $\hat{p}_1^{M_p}$ and $p_1^{N_p}$ respectively. The error between two individual labels, $l^{\text{sil}}(a, \hat{a})$, is the SNFE if one (or both) of the labels represent silence and zero otherwise. Additionally, $l^{\text{sil}}(a, \hat{a})$ is zero if one (or both) of the labels are contained in a word which represents silence, i.e. an utterance start or end word. This last check is performed because discrimination between words representing silence and other words is present in the standard word-level MBR criterion.

In Equation 7.4.1, the scalars α_w and α_p are, respectively, the word-level and phoneme-level weighting factors of Table 7.4. These weights ensure that word-end silence errors are equally influential in the cases of the phoneme-level criterion and the silence-sensitive word-level criteria.

This adjustment to the word-level MBR criterion yields a criterion which is sensitive to errors resulting from the presence or absence of word-end silence. The similarity of the phoneme-level and silence-sensitive word-level MBR criteria is measured in Section 7.6. The generalisation of phoneme-level and silence-sensitive word-level MBR-estimated acoustic models is measured in Section 7.7. These experiments provide an answer to question C of Section 7.1.

7.5 Model and state-level MBR criteria

The MBR criterion was introduced in Section 3.2.5 as the empirical risk of the classifier using the loss function $\lambda_{\text{MBR}}(h(\mathbf{o}_1^T|\theta)|\hat{w}_1^M)$ defined in Equation 7.5.1.

$$\lambda_{\text{MBR}}(h(\mathbf{o}_1^T|\theta)|\hat{w}_1^M) = \sum_{w_1^N \in \mathcal{W}} p(w_1^N|\mathbf{o}_1^T, \theta)L(w_1^N, \hat{w}_1^M) \quad (7.5.1)$$

Given that the function $L(w_1^N, \hat{w}_1^M)$ is the Levenshtein distance between two label sequences, and leaving the hypothesis space \mathcal{W} unspecified, Equation 7.5.1 is a template loss function which becomes fully specified on defining a hypothesis space. The theory of MBR criterion optimisation introduced in Section 5.1 holds for all resulting instances of the MBR criterion. Word-level MBR defines the set of word sequences as the hypothesis space while phoneme-level MBR uses the set of phoneme sequences.

In large vocabulary ASR systems, the acoustic models are often tied 3-state triphone HMMs. Section 2.3 introduced parameter-tying methods which are commonly used. The tied acoustic units of HMMs and states can be used to define MBR criteria which are informed by acoustic modelling errors. Model-level MBR deploys the set of sequences of tied HMMs as the hypothesis space. State-level MBR defines the hypothesis space as the set of sequences of tied states.

Figure 7.4 gives an example of potential behaviour differences between phoneme-level and model-level MBR. In this example the phonemic transcriptions of the reference ‘I HAVE TO LOOK AT YOU THIS YEAR’ and Hyp1, ‘I HALF TO LOOK AT CHEW THIS SHEAR’ differ by a Levenshtein distance of 3. However the model-level sequences associated with the reference and Hyp1 are identical. This is a consequence of the parameter-tying scheme which has tied the contexts ‘ae-f+t’ and ‘ae-v+t’ to the same model as well as the context pairs ‘t-y+uw’/‘t-ch+uw’ and ‘s-y+iy’/‘s-sh+iy’. Suppose that the language model probabilities of the reference and Hyp1 are identical. Then the reference and Hyp1 have the same posterior probability, regardless of the acoustic model parameters. Similarly to the scenario illustrated in Figure 7.1, it can be shown that in this case the posterior probability of the reference transcript will be reduced after phoneme-level MBR parameter re-estimation. However the model-level MBR criterion has the desirable effect of increasing the posterior probability of the reference transcript.

The following important property of the model-level MBR criterion is illustrated by this example. The model-level criterion focusses upon confusion which can be resolved via the adjustment the HMM parameters of the acoustic model and disregards confusions which are resolved via other acoustic model components such as the dictionary and the phonetic decision tree used to tie the acoustic model parameters. The state-level MBR criterion also possesses this property and is similarly motivated.

Question D of Section 7.1 has been addressed. While the above example provides theoretical motivation for the use of model and state-level MBR criteria, it is often the case that triphone model clustering is constrained such that only triphone contexts with the same centre phoneme may share the same model. In the example of Figure 7.4 such constraints are assumed to be absent.

WORD-LEVEL	Reference	I HAVE TO LOOK AT YOU THIS YEAR			
	Hyp1	I HALF TO LOOK AT CHEW THIS SHEAR			
	Hyp2	I HAVE TO LOOK AT YOU MISS YEAR			
PHONEME-LEVEL	Reference	ay hh ae v t ax l uh k ax t y uw dh ih s y iy r	Levenshtein error	Posterior	Updated posterior
	Hyp1	ay hh ae f t ax l uh k ax t ch uw dh ih s sh iy r	0	0.3	0.0
	Hyp2	ay hh ae v t ax l uh k ax t y uw m ih s y iy r	3	0.3	0.0
MODEL-LEVEL	Reference	ay1 hh1 ae1 fl t1 ax1 ll uw1 k1 ax1 tl ch1 uw1 dh1 ih1 s1 sh1 iy1 r1	0	0.3	0.5
	Hyp1	ay1 hh1 ae1 fl t1 ax1 ll uw1 k1 ax1 tl ch1 uw1 dh1 ih1 s1 sh1 iy1 r1	0	0.3	0.5
	Hyp2	ay1 hh1 ae1 fl t1 ax1 ll uw1 k1 ax1 tl ch1 uw1 m1 ih1 s1 sh1 iy1 r1	1	0.4	0.0

Figure 7.4: *Comparison of phoneme and model-level MBR criteria. In this example the phonemic transcriptions of the reference and Hyp1 differ by a Levenshtein distance of 3, while the model-level sequences associated with the reference and Hyp1 are identical. Suppose that the language model probabilities of the reference and Hyp1 are identical. Then the reference and Hyp1 have the same posterior probability, as shown in the column labelled ‘Posterior’. The phoneme-level MBR criterion achieves its minimum by adjusting the posterior probabilities as shown in the column labelled ‘Updated posterior’. Note that the posterior probability of the reference transcription is decreased to 0. Compare this with the effect of the model-level MBR criterion. Since the model sequences corresponding to the reference transcription and Hyp1 are identical, the model-level MBR criterion achieves its minimum by reducing the posterior probability of the second hypothesis to 0. Note that the posterior probability of the reference transcription is increased to 0.5.*

However, even with such parameter-tying constraints, model or state-level MBR-estimated models may exhibit behaviour different to word and phoneme-level MBR-estimated models. Figure 7.5 compares the Levenshtein error of the word, phoneme, model and state sequences corresponding to the hypotheses Hyp1 and Hyp2. Suppose that these hypotheses have equal posterior probability. To achieve its minimum, the MBR criterion places emphasis upon the reduction of the posterior probability of the hypothesis with the larger error. So the word-level MBR criterion emphasises the correction of Hyp1 while the phoneme-level criterion emphasises correction of Hyp2. The model-level MBR criterion emphasises the correction of Hyp1 while the state-level criterion emphasises correction of Hyp2. Consequently, acoustic models with different properties potentially emerge as a result of model re-estimation using these different MBR formulations.

The relative generalisation of each of the MBR formulations is measured via an experimental evaluation in Section 7.7. Before proceeding to this evaluation, the similarity of each of these criteria is measured. This analysis helps to interpret the results of Section 7.7.

			Levenshtein error
WORD-LEVEL	Reference	LET SHOUT	
	Hyp1	LETS OUT	2
	Hyp2	LOUD SHOUT	1
PHONEME-LEVEL	Reference	l eh t sh ow t	
	Hyp1	l eh t s ow t	1
	Hyp2	l ow d sh ow t	2
MODEL-LEVEL	Reference	l_1 eh_1 t_1 sh_1 ow_1 t_2	
	Hyp1	l_1 eh_1 t_2 s_1 ow_2 t_2	3
	Hyp2	l_1 ow_1 d_1 sh_1 ow_1 t_2	2
STATE-LEVEL	Reference	l1 l2 l3 eh1 eh2 eh3 t1 t2 t3 sh1 sh2 sh3 ow1 ow2 ow3 t21 t22 t23	
	Hyp1	l1 l2 l3 eh1 eh2 eh3 t11 t2 t3 s1 s2 s3 ow11 ow2 ow3 t21 t22 t23	5
	Hyp2	l1 l2 l3 ow1 ow2 ow3 d1 d2 d3 sh1 sh2 sh3 ow1 ow2 ow3 t21 t22 t23	6

Figure 7.5: Comparison of Levenshtein errors in word, phoneme, model and state-level hypothesis spaces. The MBR criterion places emphasis upon the reduction of the posterior probability of the hypothesis with the larger error.

7.6 Similarity of MBR criteria

In this section the similarity of the MBR criteria evaluated in Section 7.7 is measured. As discussed in Section 6.4, a measure of the similarity of two MBR criteria is the correlation between their associated error functions. To measure this correlation, the same subset of the training corpus used in Section 6.4 is used (approximately one hour of speech comprising 1851 utterances). As described beforehand, the reference word sequence of each utterance is aligned using the acoustic models of the second pass of the recognition system described in Section B.1 to yield word, phoneme, model and state-level reference alignments for each utterance. The same recognition system is used to generate word phoneme, model and state-level hypothesis alignments for each utterance.

The SNFE approximation corresponding to each utterance is calculated at the word, phoneme, model, state, phoneme-sensitive word and silence-sensitive word levels. Note that the acoustic parameter-tying is constrained as described in Section 7.7. The correlation coefficient between each set of errors is displayed in Table 7.3.

Error level	Word	Phoneme	Model	State	Phoneme-sensitive word
Phoneme	0.950	-	-	-	-
Model	0.932	0.962	-	-	-
State	0.931	0.963	0.998	-	-
Phoneme-sensitive word	0.993	0.954	0.933	0.933	-
Silence-sensitive word	0.984	0.958	0.933	0.936	0.979

Table 7.3: *Correlation between word, phoneme-sensitive word, silence-sensitive word, phoneme, model and state-level approximated errors of hypotheses corresponding to one hour of speech. The symmetrically normalised frame error is used to approximate errors.*

7.6.1 Discussion

Error measures with high correlations induce MBR criteria with similar behaviour. Given the information in Table 7.3, one therefore predicts that state and model-level MBR criteria (whose error measures display a relatively high correlation of 0.998) will behave similarly. This high correlation is a consequence of the highly constrained parameter-tying used in the acoustic model.

One may also expect the word-level and phoneme-sensitive word-level criteria to behave similarly, a correlation of 0.993 being observed between their associated error measures.

The correlations between sub-word (phoneme, model and state-level) and word-level errors are less than or equal to 0.950 in all cases. These are significantly weaker correlations than the correlations between any pair of sub-word-level errors, which is greater than or equal to 0.962. One may therefore expect sub-word MBR criteria to behave somewhat differently to the word-level MBR criterion. Note that, as beforehand (Section 6.4) a significant difference between two correlation coefficients is measured by the procedure prescribed in Steiger (1980) for correlation coefficients derived from the same sample.

Compare the correlation between silence-sensitive word-level errors and phoneme-level errors (0.958) with the correlation between standard word-level errors and phoneme-level errors (0.950). This is a significant difference. It is therefore concluded that errors related to word-end silence account for some of the differences between the word-level and phoneme-level MBR criteria.

The correlation between phoneme-sensitive word-level errors and phoneme-level errors (0.954) differs insignificantly from the correlation between standard word-level errors and phoneme-level errors (0.950). This demonstrates that the differing treatment of overlapping homophones and heteronyms does not significantly contribute to the differences between the word-level and phoneme-level MBR criteria.

7.7 Evaluation: sub-word MBR criteria

A large vocabulary task of meeting speech transcription is used to evaluate the impact of the different MBR criteria introduced in this chapter upon the generalisation of the

resulting acoustic models. Appendix B.2 describes the two-pass recognition system used in this evaluation. Appendix B.3 describes the MBR model estimation procedure in detail. The SNFE approximation introduced in Chapter 6 is used to approximate the Levenshtein error in all cases.

The training dataset used is a 64 hour subset of the full 104 hour training dataset described in Section B.1.5. The different MBR-estimated acoustic models are then substituted into the second recognition pass of the transcription system and evaluated. The *rt05seval*, *rt06seval* and *rt07seval* NIST evaluation datasets (see Section B.1.5 for details) are used as test data.

Note that the acoustic models used are phonetic-decision-tree-tied triphones (see Section 2.3.1), where only triphone models sharing the same centre phoneme can share states. Additionally, only states with the same position within their containing HMM can be tied (i.e. first states can only be tied with other first states and similarly for second and third states).

7.7.1 Unsmoothed MBR

Figure 7.6 displays how the average recognition WER (on the *rt05seval*, *rt06seval* and *rt07seval* datasets) yielded by the MBR-estimated models varies with each iteration of parameter estimation. Each of the displayed trajectories corresponds to a different MBR formulation and the zeroth iteration corresponds to the performance of the ML-estimated models which are used as a starting point.

Discussion

The effect of overfitting the training data is observed after two or three iterations of model re-estimation in all cases. Comparing the performance of the models yielded by the second training iteration, the word-level MBR-estimated models yield the poorest generalisation (WER of 34.5%) of all four formulations considered, while the phoneme-level MBR-estimated models yield the best generalisation (WER of 34.2%). The model and state-level criteria yield models which display a WER of 34.3%. The MPSSWE significance test reveals a significant difference between the phoneme-level and the word-level systems and also between the state-level and the word-level systems. No significant difference exists between any other pair of second-iteration systems.

This result shows that both state and phoneme-level MBR criteria provide significantly improved generalisation over word-level MBR. The experiment described in the next section tests if this improved generalisation persists when using smoothed MBR criteria.

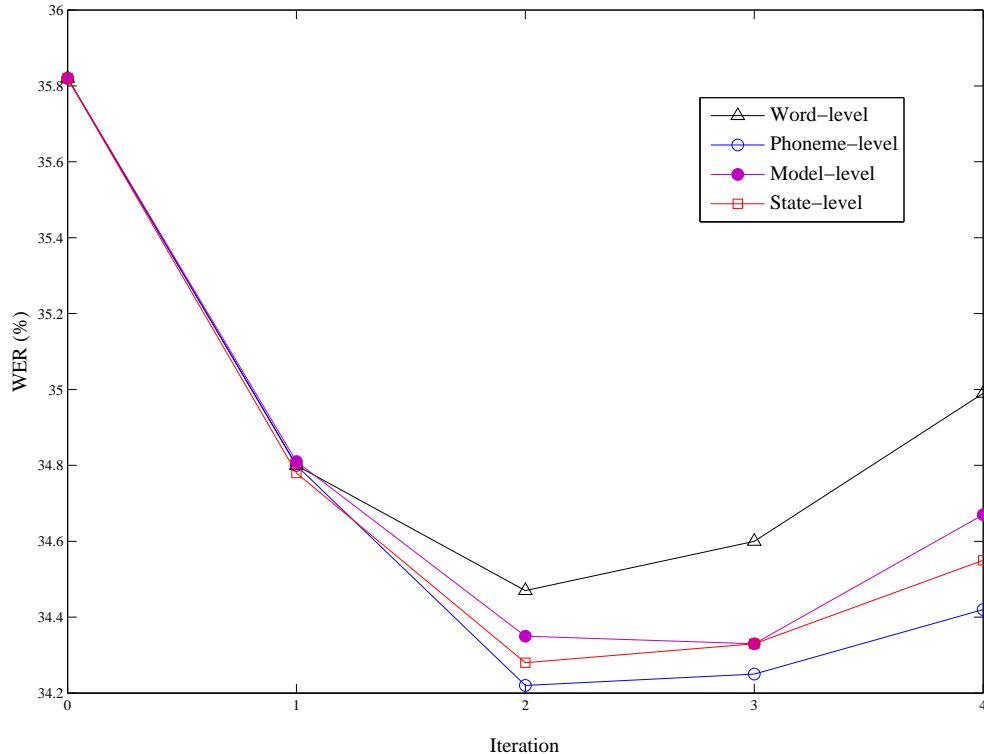


Figure 7.6: Performance of unsmoothed MBR-estimated models using different MBR formulations (rt05seval, rt06seval and rt07seval datasets). The MBR formulation is indicated in the legend.

7.7.2 I-smoothed MBR

Recall the form of the smoothed MBR criterion introduced in Equation 5.2.6 and repeated here for convenience in Equation 7.7.1.

$$R_{\text{MBR}}^S(\theta) = \frac{1}{R} \sum_{r=1}^N \sum_{w_1^N \in \mathcal{W}} p(w_1^N | \mathbf{o}_1^{T(r)}, \theta) L(w_1^N, \hat{w}_1^{M(r)}) - \log p(\theta) \quad (7.7.1)$$

Notice that the smoothed criterion is a combination of a discriminative term (the first term on the right hand side of Equation 7.7.1) and a smoothing term $\log p(\theta)$. To ensure that the influence of the smoothing term is the same for all criteria, care must be taken to ensure that the ratio of the discriminative term to the smoothing term is constant across all criteria. The value of the discriminative term before the first iteration of MBR re-estimation is given

in Table 7.4 for each MBR criterion used in the experimental work of this section. The

Criterion level	Word	Phoneme	Model	State	PS word	SS word
Criterion value	397055	892891	1075753	3251161	860852	964048
Weighting factor	2.249	1.0	0.830	0.275	1.037	0.926

Table 7.4: *MBR criterion value and weighting factor for different MBR criteria. The weighting factor for a particular criterion is the ratio the phoneme-level criterion value to the MBR criterion value. Scaling the discriminative term of the MBR criterion by its associated weighting factor ensures that I-smoothing is equally influential across all criteria. ‘PS word’ and ‘SS word’ refer to the phoneme-sensitive and silence-sensitive word-level criteria respectively.*

ratio of the phoneme-level criterion value to the MBR criterion value in Table 7.4 gives a weighting factor for each criterion, as shown in Table 7.4. Scaling the discriminative term of the MBR criterion by its associated weighting factor ensures the smoothing term is equally influential across all criteria. This is implemented by multiplying the approximate error by the appropriate weighting factor.

Using an identical experimental setup to that used in the evaluation of Section 7.7.1, multiple iterations of smoothed MBR re-estimation are performed and evaluated on the *rt05seval*, *rt06seval* and *rt07seval* datasets. An I-smoothing factor τ^I equal to 50 is used (see Section 5.2.4). Figure 7.7 compares the test set WER yielded by the MBR-estimated models after each iteration of parameter estimation. Each displayed curve corresponds to a different MBR criterion formulation, as indicated by the legend. Table 7.5 displays the WER of the models yielded after ten iterations of MBR estimation.

Criterion level	Sub (%)	Del (%)	Ins (%)	WER (%)
Word	17.4	13.5	3.1	33.9
Phoneme	17.6	13.0	3.2	33.8
Model	17.4	13.2	3.2	33.8
State	17.4	13.2	3.1	33.7
Phoneme-sensitive word	17.4	13.6	3.1	34.0
Silence-sensitive word	17.4	13.2	3.1	33.7

Table 7.5: *Performance of tenth iteration I-smoothed ($\tau^I = 50$) MBR-estimated models (*rt05seval*, *rt06seval* and *rt07seval* datasets).*

Discussion

The influence of the smoothing term in the criterion alleviates the effect of overfitting the training data. Consequently, improved generalisation is observed for all MBR criteria, in comparison to the results displayed in Figure 7.6. The average WER converges after

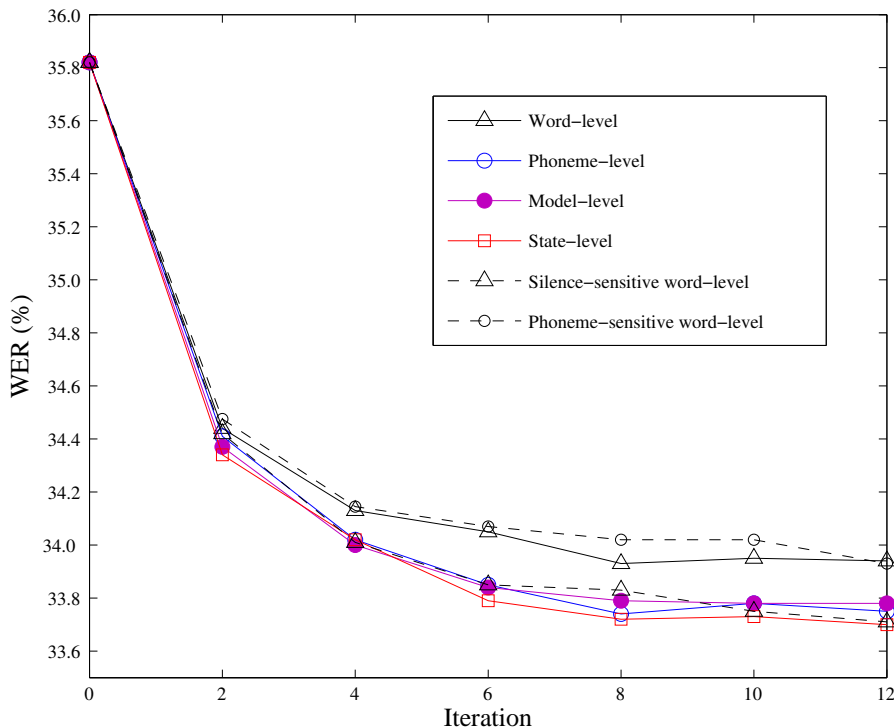


Figure 7.7: Performance of I -smoothed MBR-estimated models using different MBR formulations (*rt05seval*, *rt06seval* and *rt07seval* datasets). The MBR formulation is indicated in the legend. The I -smoothing factor τ^I is set to 50.

approximately ten training iterations. All comparisons discussed here relate to the models yielded after the tenth iteration of MBR estimation.

The MPSSWE significance test reveals a significant difference between the performance of the state-level and word-level MBR-estimated models and between the word and silence-sensitive word-level MBR-estimated models. The significance test between the phoneme-level and word-level tenth iteration MBR-estimated models reveals a significant performance difference at the 93% confidence level. The word-level and model-level MBR-estimated models yield a significant performance difference at the 94% confidence level.

No significant difference is found between the performance of the word-level and the phoneme-sensitive word-level MBR-estimated models. No significant difference is found between the performance of the sub-word MBR-estimated models (phoneme, model and state-level) and the silence-sensitive word-level MBR-estimated models. No significant performance difference is found between any pair of sub-word MBR-estimated models.

Examination of Table 7.5 shows that the word-level MBR-estimated models yield poorer generalisation mainly due to a deletion rate of 13.5% which exceeds that of the other models. In the case of the silence-sensitive word-level MBR-estimated models, a lower deletion rate of 13.2% is observed and a significantly improved overall WER (in comparison to standard word-level MBR) is yielded. From this analysis, one concludes that the high deletion rate of the word-level MBR-estimated acoustic models is due to its failure to incorporate errors related to word-end silence. Further, one concludes that the superior deletion rate and superior overall performance of the phoneme-level MBR-estimated models is attributable, at least partly, to the fact that the phoneme-level criterion respects errors related to word-end silence.

The lack of significant difference between the performance of word-level and phoneme-sensitive word-level MBR-estimated models indicates that the different treatment of overlapping homophones and heteronyms does not account for the performance difference between word and phoneme-level MBR-estimated models.

The high correlation between the model and state-level error functions, as shown in Table 7.3, explains the similar performance yielded by the resulting MBR-estimated acoustic models.

7.8 Summary and future work

This chapter has addressed the questions asked in Section 7.1. In response to question A, arguments in favour of the use of the phoneme-level MBR criterion over the word-level MBR criterion have been presented.

It has been shown that differing treatment of homophones and heteronyms of reference words account for some of the differences between the word-level and phoneme-level MBR criteria. However, with respect to question B, no evidence has been found which demonstrates that the superior generalisation of phoneme-level MBR-estimated models is attributable to this difference.

With regard to question C, it has been shown that errors related to word-end silence account for some of the differences between the word-level and phoneme-level MBR criteria. Additionally, it has been shown that the superior generalisation of phoneme-level MBR-estimated models over word-level MBR-estimated models is at least partly attributable to the inclusion of errors related to word-end silence in the case of the phoneme-level criterion.

In response to question D, novel model and state-level MBR criteria have been motivated and experimentally evaluated. With regard to question E, experimental evidence shows that significant improvements in generalisation over word-level MBR-estimated models are yielded by sub-word MBR-estimated models, both in the I-smoothed and unsmoothed scenarios.

7.8.1 Future work

No significant difference between the generalisation of the sub-word criteria defined at the phoneme, model and state-level has been recorded for the large vocabulary ASR system

considered in the experimental work presented in this chapter. Analysis of the error approximation functions reveals a high degree of correlation between the sub-word errors, leading to effectively similar MBR-estimated acoustic models. This high correlation is a consequence of the constraints placed upon the parameter-tying technique used within the acoustic model. Future research may compare the effects of the different sub-word MBR criteria when estimating acoustic models with less constrained parameter-tying schemes.

Chapter 8

Confidence-driven MBR acoustic model adaptation

8.1 Introduction

A range of acoustic model adaptation methods have been introduced in Chapter 4. This chapter concentrates upon the application of linear regression adaptation, in particular the discriminative minimum Bayes risk linear regression (MBRLR) technique, introduced in Section 4.4.1, to the task of unsupervised speaker adaptation.

Unsupervised speaker adaptation is characterised by the unavailability of the correct, or reference, transcription of the adaptation data. Additionally, the test data and the adaptation data coincide in the unsupervised adaptation scenario. A common implementation of unsupervised adaptation is to firstly estimate the transcription of the adaptation data using a recognition pass. This estimated transcription is then used in the same manner as the correct transcription in the supervised adaptation process. So unsupervised adaptation usually differs from supervised adaptation only in that the reference transcription is estimated in the former case.

The inaccuracy of the estimated transcription limits the performance of unsupervised adaptation, as witnessed by the large performance improvements when the true transcription replaces the estimated transcription in the case of both MLLR (Pitz et al. (2000)) and discriminative MBRLR adaptation (Wang and Woodland (2004)). Thus efforts have been made to constrain the adaptation to respect only correctly-transcribed labels of the estimated transcription. A label may be a word sequence, a word, or a sub-word unit of speech.

Of course, it is unknown which labels of the estimated transcription are correct. However confidence measures may be used to characterise the correctness of a label and subsequently to classify it as correctly or incorrectly-transcribed. Confidence information has previously been integrated into unsupervised MLLR adaptation (Pitz et al. (2000)) and performance improvements over standard MLLR reported. In this chapter, Objective 4 of Section 1.5 is pursued. The theory of MBR linear regression adaptation is extended to incorporate

knowledge of confidence in the reference transcription, resulting in novel confidence-driven unsupervised MBRLR adaptation techniques. Several aspects of MBRLR adaptation theory are re-formulated to integrate this confidence information, including the complexity control and I-smoothing techniques.

While these novel techniques provide the focus of this chapter, the theory, implementation and experimental results pertaining to confidence-driven MLLR are also presented to provide informative theoretical and performance comparisons with confidence-driven MBRLR.

The chapter is structured as follows. A review of previous work on confidence estimation in the domain of ASR is presented in Section 8.2. Section 8.3 explains how such confidence information is integrated into unsupervised MLLR adaptation. Section 8.4 presents the theory of confidence-driven MBRLR adaptation. Section 8.5 describes the large vocabulary recognition system used to evaluate these techniques. Evaluations of the confidence-driven MLLR and MBRLR techniques are presented in Sections 8.7 and 8.8 respectively. Analysis of the different confidence measures used is essential to the interpretation of the results of these evaluations and is therefore presented beforehand in Section 8.6. A concluding discussion and propositions for future research are found in Section 8.9.

8.2 Confidence estimation

Much research has been done to identify suitable measures of word-level confidence in large vocabulary ASR. These confidence measures are used as input features to a binary classifier which either classifies a word in the ASR transcription as correct or incorrect. Useful confidence measures enable this classification task to be performed accurately. These measures can be categorised as those which can be obtained from models used by the ASR system and those which are obtained from alternative models (not used by the ASR system). Examples of the latter type of confidence measure include the metamodells introduced in Cox and Dasmahapatra (2002). These are statistical HMMs which represent the probability of phoneme insertion, substitution and deletion. The models are trained in the same way as acoustic HMMs and results published in Cox and Dasmahapatra (2002) report some improvements over other probabilistic confidence estimation techniques based on N -best lists (Gillick et al. (1997)).

Successful examples of word-level confidence measures obtained from models used by the ASR system are acoustic stability (Finke et al. (1996)) and hypothesis density (Kemp and Schaaf (1997)). Acoustic stability tests the sensitivity of transcribed words to different LM scaling factors. A list of alternative transcriptions is used, where each member of the list is a recognised word sequence corresponding to use of a particular LM scaling factor in recognition. Using this list, the frequency of occurrence of a particular word in a particular position within the transcription is measured. Words of high frequency are assigned a relatively high confidence and low frequency words are assigned a relatively low confidence.

The alternative hypothesis density confidence measure is based on word lattices which encode the most likely alignments of an utterance. The hypothesis density of a particular

word at a particular frame t is the number of lattice arcs corresponding to this word which span frame t .

Confidence measures related to the word posterior probability have proven more successful (with regard to classification of words as correct or incorrect) than both the acoustic stability and hypothesis density features (Wessel et al. (2001)). These measures are used in the experimental work of this chapter and are introduced in detail in the following section.

8.2.1 Posterior-based confidence measures

Word lattices are used as follows to derive posterior-based confidence measures. Consider the lattice shown in Figure 8.1. Each lattice arc is marked with its associated word label and its posterior probability. For example ‘B/0.3’ means that the arc has word label ‘B’ and posterior probability of 0.3. The lattice path of maximum posterior probability is shown as the ‘MAP alignment’. For each word label of the MAP alignment, an associated confidence measure is sought. For each label X in the MAP alignment, with start time X_s and end

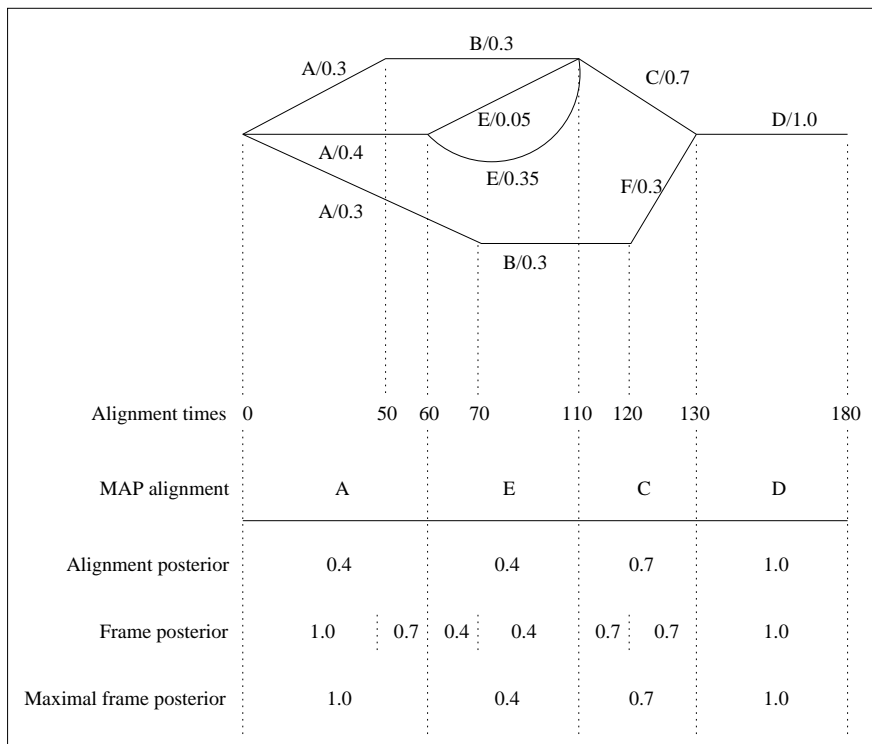


Figure 8.1: Calculation of posterior-based confidence measures from a lattice of word alignments.

time X_e , the posterior probability of this label, $p(X, X_s, X_e | \mathbf{o}_1^T, \theta)$, given the acoustic data \mathbf{o}_1^T and model parameters θ , may be calculated by summing the posterior probabilities of

all the lattice arcs with matching label and identical start and end times. This quantity is referred to as the alignment posterior. In Figure 8.1, the alignment posterior of the label ‘E’ is the sum of the arc posteriors for the arcs labelled ‘E’ since these arcs have matching start and end times.

The alignment posterior yields a somewhat poor measure of confidence (Wessel et al. (2001)), again with respect to the classification of words as correct or incorrect. This is because, typically, a lattice comprises several alignments containing the same label with slightly different start and end times. In such a situation the alignment posterior underestimates the confidence in the label since posterior probabilities are distributed across each different alignment. Notice that this is the case for label ‘A’ in Figure 8.1.

Alternative confidence measures involve the use of the frame posterior, defined as follows. Each frame defines a set of lattice arcs overlapping the frame and an overlapping label in the MAP alignment, ‘X’ say. The frame posterior is defined as the sum of the posteriors of each of the arcs overlapping frame t which have label ‘X’. In Figure 8.1, the label ‘A’ has frame posterior of 1.0 between the frames of 0 and 50 due to the contributions of each of three lattice arcs overlapping this time period.

The confidence associated with a label of the MAP alignment may be derived from the frame posteriors in several ways. The most reliable confidence measure reported in Wessel et al. (2001) is defined by choosing the maximal value of the frame posterior between the start and end frames of the label. This measure is referred to as the maximal frame posterior and displayed in Figure 8.1. To illustrate, note that label ‘A’ in the MAP alignment has frame posterior of 1.0 in the time period between 0 and 50 frames and frame posterior of 0.7 in the time period between 50 and 60 frames. So maximisation of the frame posterior in the region between the start and end frames of the label ‘A’ gives this label a maximal frame posterior of 1.0. The maximal frame posterior does not necessarily obey the sum-to-one constraints of a probability distribution. However it yields a more successful confidence measure than the alignment posterior because alignments which differ slightly from the MAP alignment also contribute to the confidence measure.

There are several other techniques which may be used to derive a confidence measure from frame posteriors. In Wessel et al. (2001), a confidence measure for the label ‘X’ is defined as the value of the frame posterior of label ‘X’ at the mid-point of the label between the start and end times. This yields a confidence measure with similar classification performance (with regard to the classification of words as correct or incorrect) as the maximal frame posterior. The geometric mean of the frame posterior over each of the frames corresponding to the label has also been effectively used as a confidence measure (Evermann and Woodland (2000)). In the experimental work of this chapter, the maximal frame posterior is used as a confidence measure since it yields consistently superior classification performance to the alternative measures studied in Wessel et al. (2001).

8.2.2 Sub-word confidence measures

The techniques described in Section 8.2.1 have been used for the computation of posterior-based word-level confidence measures from lattices. Using lattices marked with phoneme,

model and state-level alignments, these techniques may also be used to calculate phoneme, model and state-level confidence measures. Figure 8.2 depicts such a lattice consisting of two word hypotheses ‘BIG’ and ‘PIG’ of posterior probability 0.6 and 0.4 respectively. Each word hypothesis is marked with its most likely phoneme, model and state-level alignment. The vertical dashed lines represent phoneme and model boundaries while the vertical dotted lines represent state boundaries. For example, the initial phoneme in the word ‘BIG’ is ‘b’ and this corresponds to the model ‘b_2’. The word, phoneme, model and state-level frame

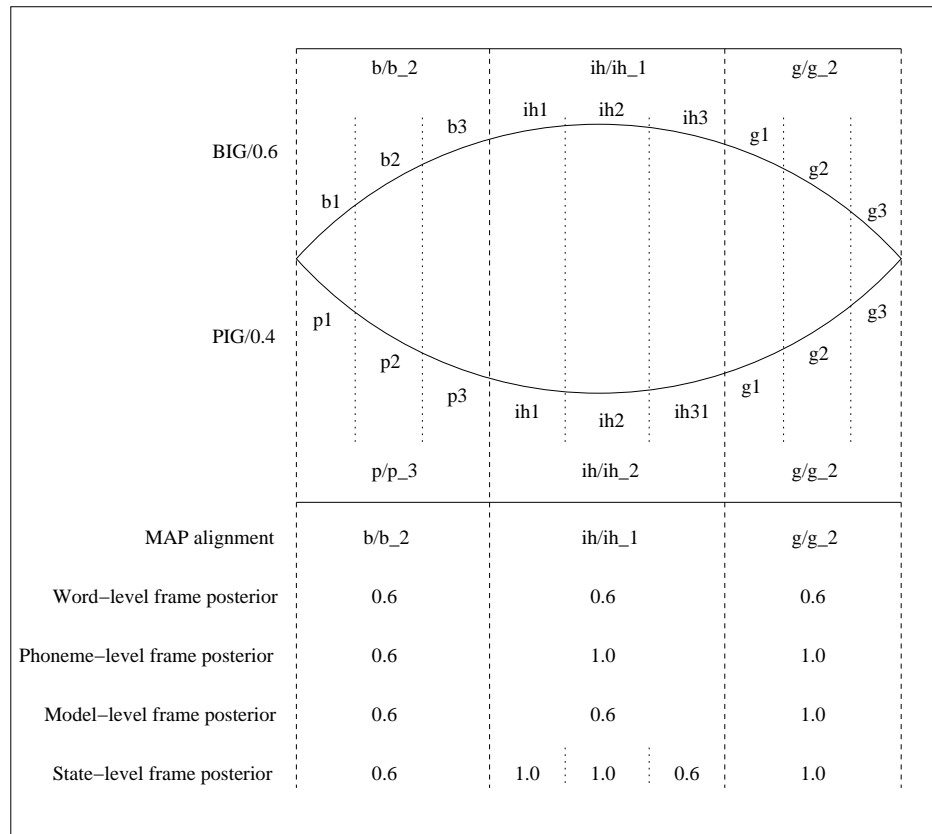


Figure 8.2: Calculation of sub-word confidence measures from a model, phoneme and state-aligned lattice.

posteriors are based upon the word, phoneme, model and state labels respectively. The maximal frame posterior at a particular level is then derived from the frame posterior at that level. The use of such sub-word confidence measures within confidence-driven speaker adaptation procedures will later be motivated. An explanation of how confidence measures have been used to inform unsupervised speaker adaptation follows.

8.3 Confidence-driven MLLR

In Pitz et al. (2000), the word-level maximal frame posterior described in Section 8.2.1 is used as a confidence measure. The MLLR adaptation procedure is then adjusted to respect only words with confidence above a certain threshold. To understand how this is implemented, Equation 8.3.1 shows how the ML criterion may be approximated as a sum of the log likelihoods of the models representing the reference transcription. The assumption used in this approximation is that alignments of the acoustic data which disagree with the MAP alignment label boundaries (i.e. label start and end times) have negligible likelihood. Note that a label may be a word, phoneme or other unit of speech.

$$\begin{aligned}\theta_{\text{ML}} &= \arg \max_{\theta} \sum_{r=1}^R \log p(\mathbf{o}_1^{T(r)} | \hat{w}_1^{M(r)}, \theta) \\ &\approx \arg \max_{\theta} \sum_{r=1}^R \sum_{k=1}^{K_r} \log p(\mathbf{o}_{s_k}^{e_k}(r) | \theta(k, r))\end{aligned}\quad (8.3.1)$$

In Equation 8.3.1, θ represents the collection of acoustic model parameters, r indexes each utterance, k indexes each label of the reference transcription, $\mathbf{o}_{s_k}^{e_k}(r)$ is the segment of the acoustic data $\mathbf{o}_1^{T(r)}$ which aligns (with respect to the MAP alignment) to the k -th label of the transcription $\hat{w}_1^{M(r)}$, and $\theta(k, r)$ denotes the acoustic models which represent this label.

8.3.1 Confidence-thresholded MLLR

The confidence-based refinement to the standard ML criterion used in Pitz et al. (2000) maximises only the likelihood of models corresponding to high-confidence words. Such a refinement may be expressed by Equation 8.3.2. Here $C(r, k)$ is the confidence associated with the k -th label of the estimated transcription of the r -th utterance and C is a predefined confidence threshold.

$$\theta_{\text{ML}}^C = \arg \max_{\theta} \sum_{r=1}^R \sum_{k: C(r, k) > C} \log p(\mathbf{o}_{s_k}^{e_k}(r) | \theta(k, r))\quad (8.3.2)$$

The maximisation of this confidence-thresholded ML criterion is implemented by disregarding statistics associated with low-confidence labels. As is clear from Figure 8.1 each frame t has an associated confidence, $C(t)$ say; the confidence associated with the label overlapping this frame in the MAP alignment. Again assuming that alignments of the acoustic data which disagree with the MAP alignment label boundaries have negligible likelihood, the statistics required for MLLR transform estimation are altered to use a confidence-adjusted occupancy, $\gamma_m^C(t | \hat{w}_1^{M(r)}, \mathbf{o}_1^{T(r)}, \theta)$, in place of the standard occupancy $\gamma_m(t | \hat{w}_1^{M(r)}, \mathbf{o}_1^{T(r)}, \theta)$. The confidence-adjusted occupancy is zero for each mixture component m at low-confidence

frames, as described by Equation 8.3.3.

$$\gamma_m^C(t|\hat{w}_1^{M(r)}, \mathbf{o}_1^{T(r)}, \theta) = \begin{cases} 0 & \text{if } C(t) < C \\ \gamma_m(t|\hat{w}_1^{M(r)}, \mathbf{o}_1^{T(r)}, \theta) & \text{otherwise} \end{cases} \quad (8.3.3)$$

The experimental work in Pitz et al. (2000) demonstrates how, with an informed choice of confidence threshold C , this technique results in significantly improved unsupervised MLLR performance when using word-level confidence measures. The task used in this publication is the transcription of conversational German speech (VERMOBIL) (Bub and Schwinn (1996)). This method is referred to as confidence-thresholded MLLR its effectiveness is evaluated in Section 8.7.1.

8.3.2 Confidence-weighted MLLR

Confidence-weighted MLLR is an alternative to the confidence-thresholded MLLR technique described above which avoids the need to specify a confidence threshold. Instead, the confidence-driven ML criterion is formulated as described by Equation 8.3.4, where the log likelihood of each label of the estimated transcription is pre-multiplied by its associated confidence. The notation of Equation 8.3.2 has been re-used.

$$\theta_{\text{ML}}^C = \arg \max_{\theta} \sum_{r=1}^R \sum_k C(r, k) \log p(\mathbf{o}_{s_k}^{e_k}(r) | \theta(k, r)) \quad (8.3.4)$$

This formulation of confidence-driven MLLR has previously been proposed in Pitz (2005). The implementation of transform estimation involves adjustment of the statistics to use a confidence-adjusted occupancy, $\gamma_m^C(t|\hat{w}_1^{M(r)}, \mathbf{o}_1^{T(r)}, \theta)$, as described by Equation 8.3.5. Again, it is assumed that alignments of the acoustic data which disagree with the MAP alignment label boundaries have negligibly small likelihood.

$$\gamma_m^C(t|\hat{w}_1^{M(r)}, \mathbf{o}_1^{T(r)}, \theta) = C(t) \gamma_m(t|\hat{w}_1^{M(r)}, \mathbf{o}_1^{T(r)}, \theta) \quad (8.3.5)$$

An evaluation of confidence-weighted MLLR is provided in Section 8.7.2.

8.3.3 Sub-word confidence-driven MLLR

It is straightforward to extend the confidence-driven approaches (both confidence-thresholded and confidence-weighted) described above to constrain the MLLR adaptation procedure to sub-word units which are transcribed with high confidence. Use of sub-word confidence information may more effectively exploit the data included in a confidence-driven adaptation scheme. This idea is illustrated in Figure 8.2.

In Figure 8.2, the initial phoneme of the word ‘BIG’ has relatively low confidence of 0.6 while the latter portion of the word has a high phoneme-level confidence of 1.0. So a phoneme-level confidence-thresholded adaptation scheme, with a confidence threshold of, for example 0.7, includes the data corresponding to the latter portion of the word ‘BIG’ and

discards the data corresponding to the initial phoneme. A word-level confidence-thresholded adaptation scheme with the same confidence threshold discards, arguably wastefully, the data corresponding to the entire word. Thus a phoneme-level confidence-driven approach may more effectively use the adaptation data.

The same argument can be made in favour of a model-level confidence-driven adaptation scheme. However an important difference exists between the model-level and phoneme-level confidence-driven adaptation schemes. This is again illustrated in Figure 8.2. Notice that while the phoneme-level confidence of the centre phoneme is 1.0, its model-level confidence of 0.6 is comparatively low. Confidence-thresholded MLLR adaptation increases the likelihood of models corresponding to regions whose associated confidence exceeds a certain threshold. So phoneme-level confidence-thresholded adaptation, with a confidence threshold of, for example 0.7, increases the likelihood of the relatively low-confidence model of the central phoneme. This is an undesirable mismatch between the adaptation criterion and the confidence measure. Such a mismatch does not occur when using model-level confidence measures; model-level confidence-thresholded MLLR adaptation only increases the likelihood of models whose associated confidence is above a certain threshold.

Notice further from Figure 8.2 that the central model ‘ih_1’ has relatively low model-level confidence of 0.6 while two of its three constituent states have a relatively high confidence of 1.0. This illustrates that state-level confidence-driven adaptation schemes could prove yet more efficient than model-based schemes in their use of adaptation data.

Analysis of the behaviour and performance of word and sub-word confidence measures is presented in Section 8.6. The relative effectiveness of these different confidence measures upon the performance of unsupervised confidence-driven MLLR is quantified experimentally in Sections 8.7.1 and 8.7.2. These experiments lend insight into the usefulness of sub-word confidence measures and provide performance benchmarks for unsupervised confidence-driven MBRLR, presented in Section 8.4.

8.3.4 Confidence-driven MLLR and complexity control

The complexity control mechanism for MLLR transform generation was explained in Section 4.3.2. An occupancy count for each regression class (i.e. the sum of the occupancies of each of the components within the regression class) is used to determine whether sufficient adaptation data exists to robustly estimate a transform corresponding to the regression class.

In the case of confidence-driven MLLR, much of the adaptation data may be disregarded. The complexity control mechanism is therefore modified to respect the sum of the confidence-adjusted occupancies (Equation 8.3.3 in the case of confidence-thresholded MLLR and Equation 8.3.5 in the case of confidence-weighted MLLR) of each of the components associated with the regression class. This adjustment to the complexity control mechanism introduces sensitivity to the volume and type of data disregarded by the confidence-driven criteria.

8.4 Confidence-driven MBRLR

Recently-published research (Wang and Woodland (2008)) has incorporated confidence information into unsupervised phoneme-level MBRLR adaptation. Word posterior-related confidence measures derived from confusion networks (Mangu et al. (1999), Evermann and Woodland (2000)) are used to weight the occupancy $\gamma_m(t|w_1^N, \mathbf{o}_1^{T(r)}, \theta)$ used in MBRLR transform estimation (Equations 5.3.4 and 5.3.5). One issue with this approach is that while the occupancies corresponding to low-confidence labels are de-emphasised, the errors assigned in accordance with these labels are unaffected. De-emphasis of the occupancy associated with low-confidence labels does not address the effect of the errors assigned in accordance with these labels. This is an important limitation, since, when optimising the MBR criterion, the error assigned to each label affects the statistics associated with all other labels.

This limitation is addressed by the alternative approach to confidence-driven MBRLR proposed here. The occupancies used in MBRLR transform estimation remain unaffected, but the error assigned in accordance with low-confidence labels is adjusted. A confidence-driven MBR criterion is firstly defined and subsequently the MBRLR transforms are estimated to optimise this criterion. The confidence-driven MBR criterion is firstly explained.

The ML criterion uses the reference transcription to define the models whose likelihood is maximised with respect to the training or adaptation data. In the case of the MBR criterion the role of the reference transcription is somewhat different, namely to define the error of each member of the set of competing hypotheses. So, unlike the refinement to the ML criterion described in Section 8.3, the confidence-based refinement to MBR adaptation proposed here does not directly adjust the occupancies corresponding to low-confidence labels of the reference transcription. Instead, errors assigned with respect to low-confidence labels of the reference transcription are deemed unreliable and either disregarded or de-emphasised. Firstly, a general form of confidence-driven MBR criterion is defined by Equation 8.4.1.

$$R_{\text{MBR}}^C(\theta) = \frac{1}{R} \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} p(w_1^N | \mathbf{o}_1^{T(r)}, \theta) L^C(w_1^N, \hat{w}_1^{M(r)}) \quad (8.4.1)$$

Here $L^C(w_1^N, \hat{w}_1^{M(r)})$ is a confidence-sensitive error between the label sequences w_1^N and $\hat{w}_1^{M(r)}$. This error function is a modification to the Levenshtein error approximation, designed to incorporate knowledge of confidence associated with the reference transcription labels. Comparing Equation 8.4.1 with the standard MBR criterion given by Equation 3.2.14, it is clear that the only difference is the use of this confidence-sensitive error function $L^C(w_1^N, \hat{w}_1^{M(r)})$. Since confidence-driven MBR amounts to a refinement of the error function, confidence-driven MBRLR transforms are estimated using the same theory as standard MBRLR, presented in Section 5.3. The only difference is that the confidence-sensitive error replaces the standard error approximation in the MBRLR transform estimation formulae.

Two confidence-driven MBRLR techniques are proposed. The first technique is referred

to as confidence-thresholded MBRLR while the second technique is referred to as confidence-weighted MBRLR. An experimental evaluation of both instances of confidence-driven unsupervised MBRLR adaptation is presented in Section 8.8. Note that confidence-thresholded MBRLR was first introduced in Gibson and Hain (2007), where the results of some initial experiments are reported.

8.4.1 Confidence-thresholded MBRLR

Confidence-thresholded MBRLR deploys a modification to the Levenshtein error approximation as illustrated in the example of Figure 8.3. Section A of Figure 8.3 shows an alignment of an estimated reference transcription and a hypothesis alignment. The symmetrically normalised frame error (Section 6.3.1) of the hypothesis is 2, the sum of the SNFE for each aligned hypothesis label. Section B of Figure 8.3 shows the confidence associated with each label of the estimated reference transcription.

A	Reference	A	B	C
	Hypothesis	A	D	E
	Length (frames)	80	40	50
	Frame error	0	40	50
	Normalisation factor	80	40	50
	Symmetrically normalised frame error	0.0	1.0	1.0
B	Confidence	0.8	0.2	0.9
	Confidence-thresholded frame error	0	0	50
	Confidence-thresholded error	0.0	0.0	1.0

Figure 8.3: (A) Standard and (B) confidence-thresholded error approximations. A confidence threshold of 0.5 is used in this example.

The confidence-thresholded frame error is a modified version of the frame error which assigns errors only with respect to high-confidence (i.e. above a specified confidence threshold) labels of the reference alignment. More precisely, for each segment of the hypothesis alignment, the confidence-thresholded frame error is zero if the segment overlaps with a low-confidence label (i.e. below a specified threshold) of the reference alignment and equal to the standard frame error otherwise. The confidence-thresholded error for each hypothesis segment is then the normalised confidence-thresholded frame error, where the normalisation

factor is the length of the shorter of the overlapping subsegments, as described in Section 6.3.1. The overall confidence-thresholded error for the hypothesis is then the sum of the confidence-thresholded error over each segment.

In the example of Figure 8.3, the confidence threshold is defined to be 0.5. The second reference label ‘B’ of the estimated reference is therefore deemed unreliable while the other labels are deemed reliable. Therefore the confidence-thresholded frame error is zero for segments overlapping with the second reference label. The overall confidence-thresholded error is 1 and the error incurred with respect to the second reference label is disregarded. Modifying the error in this way thus reduces the impact of errors associated with low-confidence labels of the estimated reference transcription.

Recall that the symmetrically normalised frame error approximation to the Levenshtein error is expressed by Equation 6.3.1. Reusing the notation of Equation 6.3.1 ($\hat{\mathcal{A}}_r$ represents the set of aligned labels in the alignment of reference sequence $\hat{w}_1^{M(r)}$ and $l(a, \hat{a})$ is the symmetrically normalised frame error between the aligned labels a and \hat{a}) the confidence-thresholded error $L^C(w_1^N, \hat{w}_1^{M(r)})$ is expressed by Equation 8.4.2, where \mathcal{A}_C is the set of aligned hypothesis labels which overlap only with reference labels of confidence greater than some threshold C .

$$L^C(w_1^N, \hat{w}_1^{M(r)}) = \sum_{\hat{a} \in \hat{\mathcal{A}}_r} \sum_{a \in \mathcal{A}_C} l(a, \hat{a}) \quad (8.4.2)$$

An evaluation of confidence-thresholded MBRLR is found in Section 8.8.2.

8.4.2 Confidence-weighted MBRLR

Confidence-weighted MBRLR is an alternative to the use of a threshold to decide which labels of the reference alignment to disregard. Instead, the error function uses confidence to weight the error derived from each aligned reference label. This is expressed by Equation 8.4.3, where the notation of Equation 8.4.2 has been reused and $C(\hat{a})$ is the confidence assigned to label \hat{a} .

$$L^C(w_1^N, \hat{w}_1^{M(r)}) = \sum_{\hat{a} \in \hat{\mathcal{A}}_r} \sum_{a \in \mathcal{A}} l(a, \hat{a}) C(\hat{a}) \quad (8.4.3)$$

The need to specify an additional threshold parameter is avoided when using confidence-weighted MBRLR. It may be also argued that use of the confidence-weighted technique does not wastefully disregard low-confidence labels of the reference transcription, opting instead to de-emphasise their impact in proportion to their associated confidence. The performance yielded by confidence-weighted MBRLR-adapted models is compared to that yielded by confidence-thresholded MBRLR-adapted models in Section 8.8.4.

8.4.3 Sub-word confidence-driven MBRLR

In Chapter 7, word and sub-word MBR criteria were introduced and compared with regard to acoustic model estimation. These word and sub-word MBR criteria may also be used for the purposes of acoustic model adaptation.

Further, confidence-driven (both thresholded and weighted) MBR criteria can be formulated at the word and sub-word levels. In the experimental work of this chapter, the confidence measure and hypothesis space correspond when using confidence-driven MBRLR, so, for example, a word-level confidence measure is used in conjunction with the word-level MBR criterion. Note that it is possible to use, for example, a state-level confidence measure in conjunction with a phoneme-level MBR criterion. However additional motivation is required to argue in favour of such a configuration.

The arguments presented in Section 8.3 in favour of use of sub-word confidence measures for confidence-driven MLLR apply also to confidence-driven MBRLR. Since, for example, use of a word-level confidence measure lacks the ability to identify high-confidence sub-word units within low-confidence words, it is conceivable that word-level confidence-driven MBRLR is a suboptimal configuration. Constraint of the adaptation procedure to errors corresponding to high-confidence sub-word labels may more efficiently exploit the available adaptation data.

The confidence-driven MBRLR techniques described above naturally accommodate use of confidence measures at either the word or sub-word levels. In the case of confidence-thresholded MBRLR, a comparison between word and sub-word criteria is found in Section 8.8.3. A similar comparison is found in Section 8.8.4 in the case of confidence-weighted MBRLR.

8.4.4 Confidence-driven MBRLR and generalisation

Consider the issue of generalisation with regard to the unsupervised adaptation scenario. Since the adaptation data and the test data coincide, one might reasonably argue that the question of generalisation does not apply to this situation. However, when using confidence-based refinements to unsupervised adaptation, the issue of generalisation becomes relevant for the following reason.

When using confidence-driven adaptation, the adaptation data is effectively subdivided into high-confidence and low-confidence subsets. The confidence-driven adaptation procedure is designed to respect, or emphasise, information learnt from the high-confidence adaptation data while the low-confidence adaptation data is disregarded or de-emphasised. The evaluation tests how well the adapted acoustic models perform upon both the low-confidence and high-confidence subsets of the adaptation data. This evaluation is therefore a measure of how well the models adapted using the high-confidence data generalise to the low-confidence data.

The experimental results of Chapter 7 have shown that significantly superior generalisation is provided by sub-word MBR criteria. With this result in mind, and the relevance of generalisation to the unsupervised confidence-driven MBRLR scenario, one can hypothesise that sub-word level confidence-driven MBRLR will display superior generalisation to word-level confidence-driven MBRLR. This hypothesis is experimentally tested in Section 8.8.5.

8.4.5 Confidence-driven I-smoothing for MBRLR

It has been shown in the experiments described in Section 7.7 that I-smoothing can improve the generalisation of MBR-estimated acoustic models. Since the issue of generalisation is relevant to confidence-driven MBRLR adaptation, as explained in Section 8.4.4, one may hypothesise that unsupervised confidence-driven MBRLR adaptation can also benefit from the use of I-smoothing.

The question of how to formulate the I-smoothing prior distribution in the case of confidence-driven MBRLR arises. It is inconsistent to use confidence information to influence the MBR criterion but to ignore this confidence information when specifying a prior for the transform \mathbf{W} . In this work, a confidence-adjusted prior $p(\mathbf{W}|C)$, of the form described by Equation 8.4.4, is used. This prior is similar to the standard MBRLR prior (Equation 5.3.6) with the exception that the confidence adjusted occupancies (Equation 8.3.3) replace the standard occupancies. This prior distribution is maximal at the confidence-driven ML estimate of the transform \mathbf{W} .

$$\log p(\mathbf{W}|C) = \frac{\tau}{2} \sum_{m \in \mathcal{R}(s)} \sum_r \sum_{t=1}^{T(r)} \gamma_m^C(t|\hat{w}_1^{M(r)}, \mathbf{o}_1^{T(r)}, \theta) (\mathbf{o}_t(r) - \mathbf{W}\boldsymbol{\xi}_m)^\top \mathbf{C}_m^{-1} (\mathbf{o}_t(r) - \mathbf{W}\boldsymbol{\xi}_m) + k \quad (8.4.4)$$

The I-smoothed confidence-thresholded MBR criterion is formulated by subtracting the prior defined above (Equation 8.4.4) from the confidence-thresholded MBR criterion (Equation 8.4.1), as specified by Equation 8.4.5. The scalar τ determines the influence of the prior term.

$$R_{\text{MBR}}^C(\theta) = \frac{1}{R} \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} p(w_1^N | \mathbf{o}_1^{T(r)}, \theta) L^C(w_1^N, \hat{w}_1^{M(r)}) - \log p(\mathbf{W}|C) \quad (8.4.5)$$

Optimisation of this criterion requires that the reference labels which are deemed as low-confidence (with respect to the threshold C) are ignored both for the purpose of error computation (as explained in Section 8.4) and for the purpose of accumulation of the statistics used to smooth the transform estimation. The latter accumulation process is implemented by using confidence-adjusted occupancies, as explained in Section 8.3. The experiments described in Section 8.8.6 evaluate the performance of I-smoothed confidence-driven MBRLR.

8.4.6 Confidence-driven complexity control for MBRLR

The complexity control mechanism for MBRLR transform generation is explained in Section 5.3.2. Essentially, the same complexity control formalism used for MLLR is used for the purposes of MBRLR transform generation.

In the case of confidence-driven MBRLR, errors corresponding to much of the data may be de-emphasised or disregarded. To introduce sensitivity to the volume and type of de-emphasised data, the confidence-adjusted occupancies are used to inform the complexity control framework. This is identical to the complexity control procedure for confidence-driven MLLR described in Section 8.3.4.

8.4.7 Summary

In this section, two novel confidence-driven MBR criteria have been introduced. The implementation of confidence-driven MBRLR speaker adaptation with respect to these criteria has been explained. The use of sub-word confidence information within this framework has been motivated. Adjusted forms of confidence-driven I-smoothing and complexity control, compatible with confidence-driven MBRLR, have been explained.

The remainder of this chapter focusses on the evaluation of the confidence-driven MBRLR techniques introduced in this section. In addition to a basic evaluation, further hypotheses with regard to the generalisation of sub-word criterion formulations (Section 8.4.4) and I-smoothing (Section 8.4.5) are addressed experimentally. The following section describes the evaluation system used.

8.5 Evaluation system

A system based upon the IHM AMI meeting speech transcription system (Hain et al. (2005a), Hain et al. (2005b)) is used for evaluation purposes. The details of the acoustic and language models used, the features, and the first and second pass processes are provided in Section B.1. The experiments described in this chapter employ only the third-pass speaker adaptation procedure, illustrated in Figure 8.4.

The acoustic models used in the second pass are adapted using linear regression (MLLR or MBRLR) and the second pass transcription is used as the reference transcription. The adaptation process alters only the means of the Gaussian mixtures of the acoustic model. Two regression classes are used for adaptation, one corresponding to speech models and one for non-speech models. An occupancy threshold of 1000.0 and full transform matrices are used, preliminary experiments having indicated the suitability of such a configuration.

In the case of MBRLR adaptation, a lattice generation process is required. Phoneme, model and state-marked lattices are generated prior to MBRLR adaptation using a bigram language model (derived from the trigram used in recognition) and the unadapted second pass acoustic models. The lattices are subsequently pruned to a maximum density of 500.0 arcs per second to reduce the computational cost of the MBR criterion optimisation process, retaining only the lattice paths of highest posterior probability.

When using confidence-driven MLLR or MBRLR, these pruned lattices and the second pass transcription are used as input to a confidence estimation procedure to calculate the maximal frame posterior confidence measures described in Section 8.2. These measures are subsequently used as input to confidence-driven adaptation procedures.

The adapted acoustic models are evaluated via a recognition pass identical in configuration to the second pass recognition process, using a language model scaling factor of 14.0. The *rt06seval* and *rt07seval* test datasets (see Section B.1.5) are used in the evaluations reported in this chapter.

Before reporting the results of the experimental evaluation of confidence-driven MLLR and MBRLR techniques, in Sections 8.7 and 8.8 respectively, an evaluation and analysis of the confidence measures used in these experiments is presented. This analysis is essential

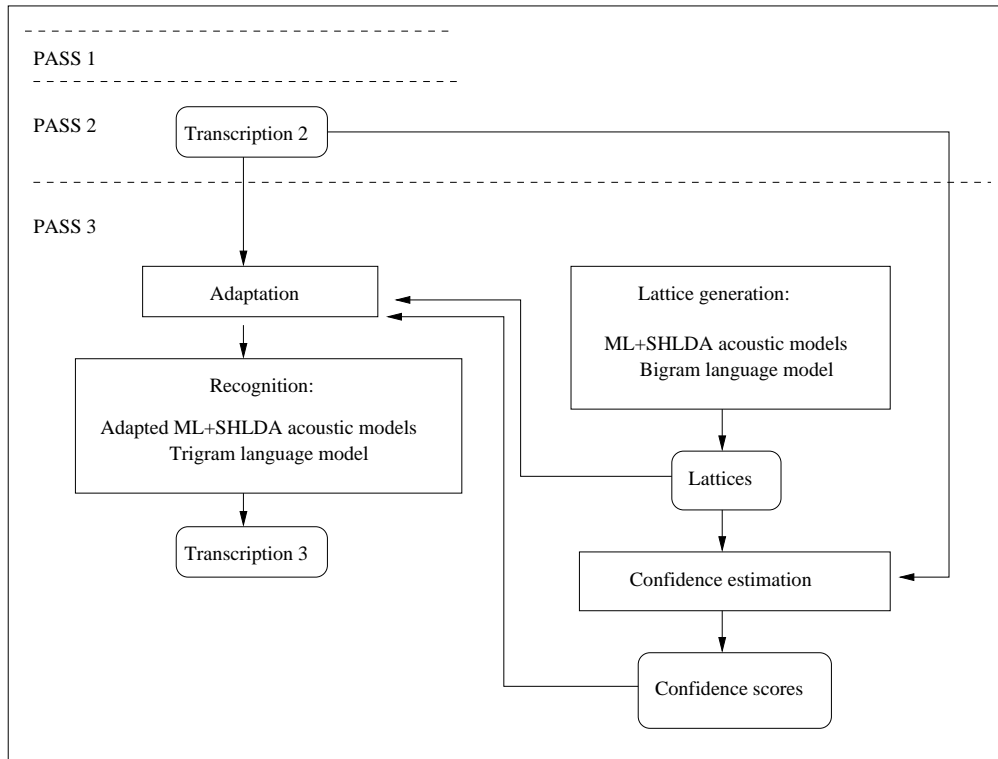


Figure 8.4: *Evaluation system for adaptation experiments. The third pass adaptation stage is evaluated.*

to the interpretation of the results of the evaluation of the confidence-driven adaptation techniques.

8.6 Evaluation: confidence measures

The *rt06seval* and *rt07seval* datasets are processed as shown in Figure 8.4. The initial two recognition passes produce a transcription of the data. This transcription is then aligned at the word, phoneme, model and state levels to give the MAP alignment at each level. A lattice generation process, as illustrated in Figure 8.4, generates phoneme, model and state-marked lattices. The maximal frame posterior confidence measures at the word, phoneme, model and state level are then calculated from these lattices and the MAP alignment at the appropriate level (as detailed in Section 8.2). Note that acoustic probability scaling is used when calculating lattice arc posterior probabilities to achieve a reasonable distribution of arc posterior probabilities (Wessel et al. (2001)). The acoustic scale factor is $\frac{1}{14}$, the inverse of the language model scale factor used in the second recognition pass. This factor is used

to scale the acoustic likelihoods associated with each lattice arc.

Categorisation of a particular frame of an alignment as correctly or incorrectly labelled defines a binary classification task. Given a confidence threshold and a frame, a confidence measure defines a binary classifier which either classifies this frame as correctly labelled (if the confidence of the label spanning this frame exceeds the threshold) or incorrectly labelled (otherwise). This section evaluates the performance and analyses the behaviour of these simple threshold-driven classifiers with respect to four confidence measures of interest; the maximal frame posterior at the word, phoneme, model and state levels.

8.6.1 Performance evaluation

The performance of the threshold-driven classifier defined above is measured by comparing the decision made by a classifier at each frame (i.e. to classify the frame as correctly or incorrectly labelled) with the correct decision. The correct decision is defined via the word, phoneme, model or state-level alignment of the correct word sequence. This alignment is referred to as the correct or reference alignment. The correct decision is to classify a frame as correct if the label of the MAP alignment agrees with the label at this frame in the corresponding correct alignment, and to classify the label as incorrect otherwise. So, e.g. the labels of the word-level MAP alignment and the word-level correct alignment define the correct decision at the word-level, while the labels of the state-level MAP alignment and the state-level correct alignment define the correct decision at the state-level. The performance of the classifier associated with a particular level is then measured in terms of how many frames are correctly classified, with respect to the correct decision at that level.

The performance of the classifiers induced by the different maximal frame posterior confidence measures is illustrated by the detection error tradeoff (DET) (Martin et al. (1997)) curves of Figure 8.5. A false alarm is defined as a frame which is misclassified as correct while a miss is defined as a frame which is misclassified as incorrect. Each datapoint corresponds to the performance of a classifier at a particular confidence threshold. The confidence thresholds range from 0.99, corresponding to high miss rates, to 0.1, corresponding to high false alarm rates. The horizontal axis measures the number of frames for which a false alarm occurs, normalised by the total number of incorrect frames (frames at which the label in the MAP alignment differs from the reference alignment). The vertical axis measures the number of frames for which a miss occurs, normalised by the total number of correct frames (frames at which the label in the MAP alignment agrees with the reference alignment label). Figure 8.6 displays these miss and false alarm rates as a function of the confidence threshold. The *rt07seval* dataset is used in this evaluation.

Discussion

Only small classification performance differences are yielded by the classifiers corresponding to the four confidence measures: the maximal frame posterior at the word, phoneme, model and state-level. Each classifier yields an equal error rate (the point where the miss and false alarm rates are equal) in the range of 24.5% to 25.8%. It is difficult to apply a significance test, for example McNemar's test (McNemar (1947)), to these results since a different

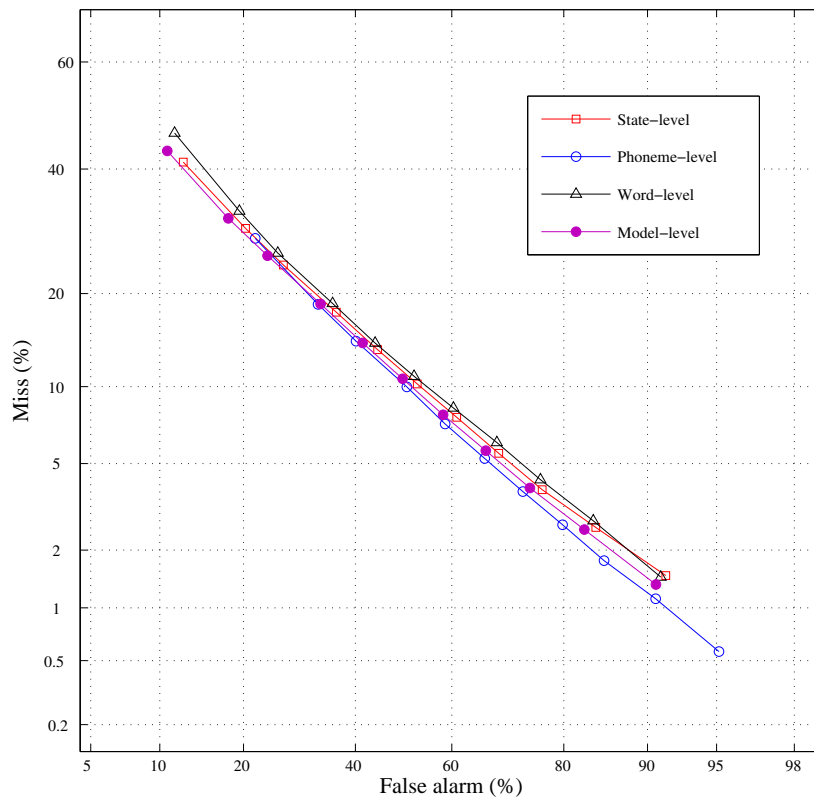


Figure 8.5: Performance of classifiers corresponding to word, phoneme, model and state-level maximal frame posterior confidence measures (*rt07seval* dataset).

classification task is associated with each classifier. One may however conclude that, given the evidence presented, relatively minor classification performance differences are displayed by the threshold-driven classifiers derived from the four different posterior-based measures.

Analysis is now presented to lend insight into the behaviour of the classifiers induced by the different confidence measures evaluated here. This analysis is used to interpret the results of the evaluation of the confidence-driven adaptation techniques, presented in Sections 8.7 and 8.8.

8.6.2 Analysis

Figure 8.7 plots the fraction of frames which are retained (i.e. deemed as correctly labelled) as a function of the confidence threshold, using the *rt06seval* and *rt07seval* datasets. Each

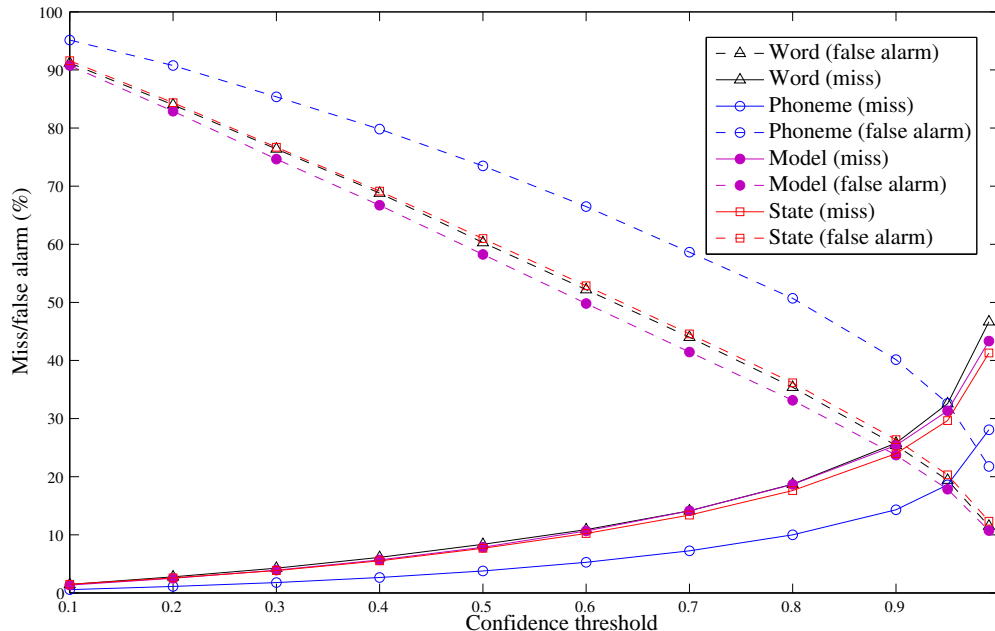


Figure 8.6: Miss and false alarm rates of classifiers corresponding to word, phoneme, model and state-level maximal frame posterior confidence measures (*rt07seval* dataset).

curve corresponds to the threshold-driven classifier derived from the maximal frame posterior confidence measure indicated by the legend. Figure 8.8 gives an example of the maximal frame posterior confidence measures associated with a small segment of speech in the *rt07seval* dataset.

It is clear from Figure 8.7 that use of the classifier induced by the phoneme-level confidence measure retains more adaptation data than the classifiers induced by the other confidence measures. This is true at all confidence thresholds.

Table 8.1 provides some analysis of the differences between the classifiers corresponding to the word and phoneme-level maximal frame posterior at the confidence threshold of 0.7. A reasonably high level of agreement is observed. The classifiers agree on whether to retain or dismiss (i.e. deem as incorrectly labelled) a frame of data for 85.07% of the frames at this threshold (the sum of the diagonal entries of Table 8.1). The main disagreement (13.85% of all frames) is due to frames which are retained by the phoneme-level classifier but dismissed by the word-level classifier at a threshold of 0.7. This is due to the effect hypothesised in Figure 8.2 and observed in Figure 8.8; the phoneme-level confidence of a phoneme in the MAP alignment is often greater than the word-level confidence of its containing word. A smaller percentage (1.08%) of frames are retained by the word-level

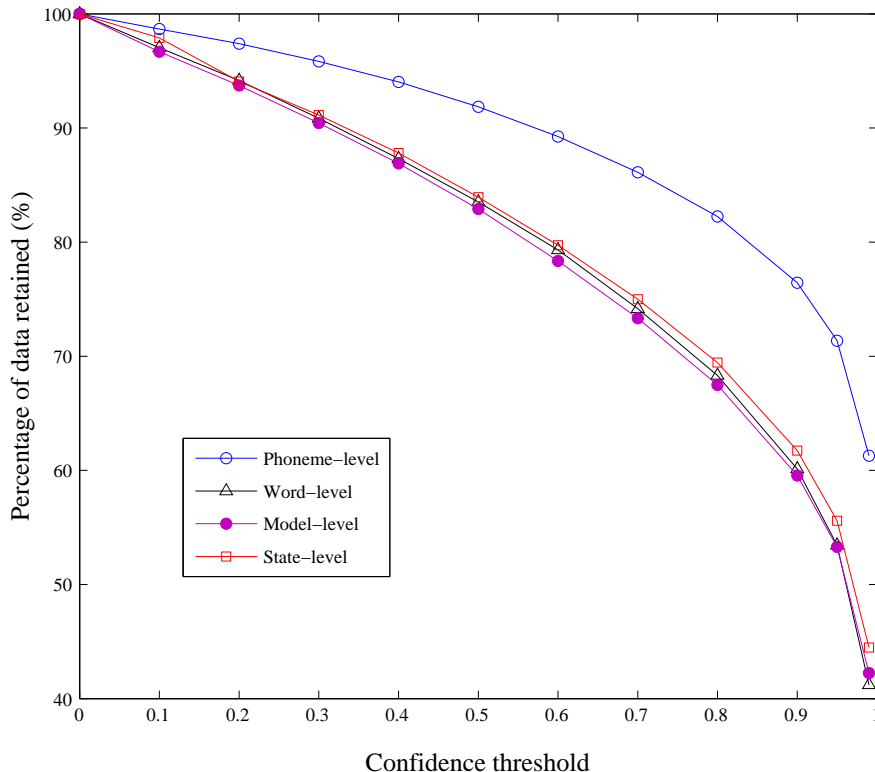


Figure 8.7: *Data retained as a function of confidence threshold (rt06seval and rt07seval datasets).*

classifier but dismissed by the phoneme-level classifier. On inspection, these frames are dismissed by the phoneme-level classifier as a consequence of differences between the within-word phoneme-level alignments of the MAP alignment and the phoneme-marked lattice.

Figure 8.7 shows that the classifiers corresponding to the word, model and state-level maximal frame posterior retain a similar volume of data at all confidence thresholds. Despite similar volumes of data being retained in the case of the model and word-level confidence measures, the frames of retained and dismissed data are somewhat complementary, as shown in Table 8.2.

The classifier corresponding to the word-level confidence measure retains 6.75% of the frames whose model-level confidence is below 0.7. This effect is due to the exclusion of data corresponding to models which display a lower model-level confidence than the word-level confidence of their containing word. This effect occurs because a word appears in several different contexts within a lattice. These different contexts correspond with different models

Reference (word)	I			MAY						NOT																						
MAP alignment (word)	I			MEAN						NOT																						
Confidence (word)	.943			.888						.998																						
Reference (phoneme)	ay			m		ey		n																								
MAP alignment (phoneme)	ay			m		iy		n		n																						
Confidence (phoneme)	.949			1.0		.922		.999		1.0																						
Reference (model)	sil-ay+m			ay-m+ey		m-ey+n		iy-n+aa																								
MAP alignment (model)	sil-ay+m			ay-m+iy		m+iy+n		iy-n+em		ng-n+aa																						
Confidence (model)	.948			.889		.921		.887		.889																						
Reference (state)	ay218	ay337	ay419	m238	m327	m439	iy233	iy327	iy416	n247	n344	n443																				
MAP alignment (state)	ay218	ay337	ay419	m239	m332	m443	iy234	iy320	iy429	n226	n324	n411	n28	n336	n442																	
Confidence (state)	.948	.948	.948	.889	.896	.885	.922	.922	.921	.887	.887	.888	.889	.889	.891																	
Frame	206		211		213		215		217		220		222		224		225		226		227		228		229		231		232		234	

Figure 8.8: Maximal frame posterior confidence measures at the word, phoneme, model and state levels for a sample of speech in the *rt07seval* dataset.

due to the presence of different context-sensitive triphone models at the start and end of the word. So while a word may be predominantly present in a lattice at a frame near the start or end of the word, the corresponding model at this frame may be less dominant. The result is that start or end models occasionally have lower model-level confidence than the word-level confidence of the containing word. This effect is observed in Figure 8.8, the first model of the word ‘NOT’ having a model-level confidence of 0.889, while the containing word has word-level confidence of 0.998.

Notice also that 6.74% of frames in the datasets considered have model-level confidence above 0.7, but word-level confidence below this threshold. This is due to the model-level confidence of a model in the MAP alignment occasionally being greater than the word-level confidence of its containing word, an effect illustrated in Figure 8.2.

A comparatively high level of agreement (95.71% of all frames) is found between the classifiers derived from the state and model-level confidence measures at the threshold of 0.7, as shown in Table 8.3. The classifier corresponding to the state-level confidence measure retains slightly more data (2.98% at this threshold) than the model-level classifier.

	frames (%)	
	retained at word-level	dismissed at word-level
retained at phoneme-level	72.26	13.85
dismissed at phoneme-level	1.08	12.81

Table 8.1: *Differences between classifiers induced by the word and phoneme-level maximal frame posterior confidence measures at a threshold of 0.7 (rt06seval and rt07seval datasets).*

	frames (%)	
	retained at word-level	dismissed at word-level
retained at model-level	66.59	6.74
dismissed at model-level	6.75	19.91

Table 8.2: *Differences between classifiers induced by the word and model-level maximal frame posterior confidence measures at a threshold of 0.7 (rt06seval and rt07seval datasets).*

This is due to the inclusion of data corresponding to states which display a higher state-level confidence than the model-level confidence of their containing model. Again this phenomenon is hypothesised in Figure 8.2 and observed in Figure 8.8. Due to alignment differences between the state-level MAP and lattice alignments there are a small number (1.32% at this threshold) of frames which are retained by the model-level classifier but dismissed by the state-level classifier.

	frames (%)	
	retained at state-level	dismissed at state-level
retained at model-level	72.02	1.32
dismissed at model-level	2.98	23.67

Table 8.3: *Differences between classifiers induced by the state and model-level maximal frame posterior confidence measures at a threshold of 0.7 (rt06seval and rt07seval datasets).*

While the analysis of this section has compared the classifiers induced by the different confidence measures at a threshold of 0.7, this analysis characterises the different behaviour of these classifiers in general, i.e. at all confidence thresholds.

8.7 Evaluation: confidence-driven MLLR

This section presents the results of an experimental evaluation of unsupervised confidence-driven MLLR adaptation. Section 8.7.1 evaluates the performance of word and sub-word confidence-thresholded MLLR. The results are compared with the performance of confidence-weighted MLLR in Section 8.7.2. Lastly, the optimal performance of the confidence-driven MLLR technique is measured in Section 8.7.3 by using ideal confidence measures. The performance of confidence-driven MLLR adaptation serves as a benchmark for comparison with confidence-driven MBRLR performance.

The multi-pass recognition system described in Section 8.5 is used, where the third-pass adaptation step is standard MLLR or confidence-driven MLLR. Four adaptation iterations of adaptation are deployed in all experiments, no significant changes in WER being observed with a larger number of iterations.

8.7.1 Experiment 1: Confidence-thresholded MLLR

Figure 8.9 displays the WER of the confidence-thresholded MLLR-adapted models as a function of the confidence threshold. The *rt06seval* and *rt07seval* datasets are used as test data. Each curve corresponds to a maximal frame posterior confidence measure as indicated by the legend. The unadapted models yield a WER of 35.8% and standard unsupervised MLLR corresponds to a confidence threshold of 0.0, a WER of 33.9%.

The curves of Figure 8.9 describe a tradeoff between the amount of adaptation data retained and the accuracy of the transcription of the retained data. At thresholds above 0.95 the volume of retained adaptation data quickly decreases and consequently the performance of the confidence-based adaptation scheme is compromised. At thresholds nearer 0.0 the confidence-based adaptation scheme uses almost all the adaptation data and the adapted system yields performance similar to that of standard MLLR. Each curve achieves a minimum within the range of 0.7 to 0.95, corresponding to the optimal confidence threshold for the associated confidence measure.

While the best performance is achieved using the phoneme-based confidence measure at a threshold of 0.9, no significant difference is found between this system and the word, model and state-level confidence-thresholded systems at their optimal thresholds of 0.9, 0.7 and 0.7 respectively. The optimal confidence threshold corresponds with a WER of 33.7% in the case of all four confidence measures. This constitutes a significant reduction over the WER of standard MLLR in all cases.

8.7.2 Experiment 2: Confidence-weighted MLLR

Table 8.4 displays the performance of the confidence-weighted MLLR-adapted models for the *rt06seval* and *rt07seval* datasets. The initial row details the performance of standard MLLR adaptation and the remaining rows correspond to the maximal frame posterior confidence measures indicated in the second column. Small, consistent improvements over standard MLLR are yielded by confidence-weighted MLLR when using the maximal frame posterior confidence measure at the word, phoneme, model and state levels. Significance

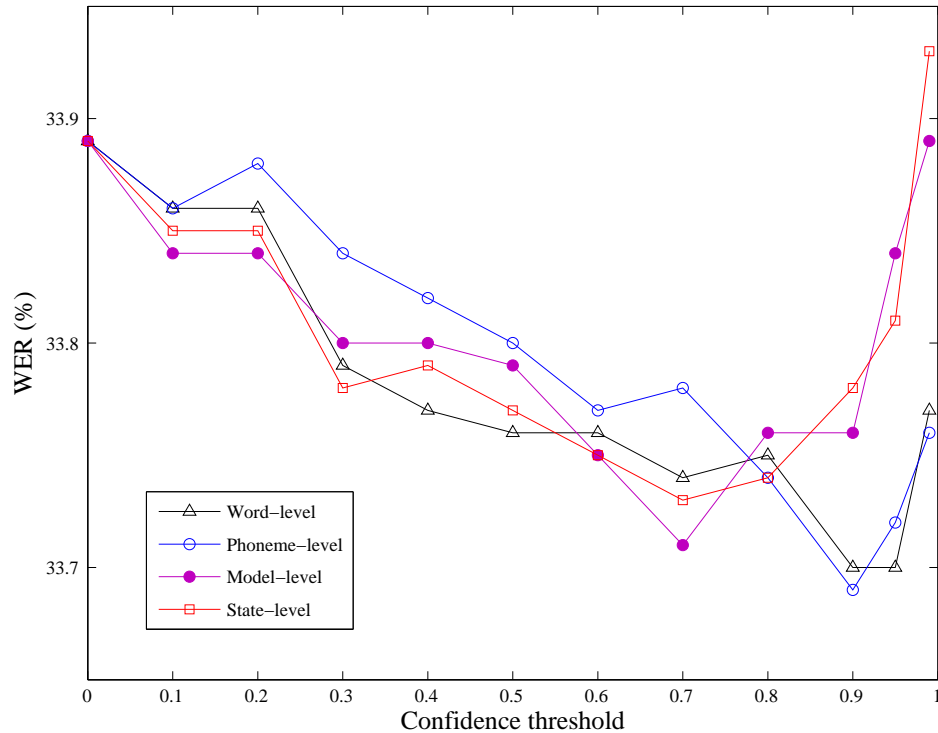


Figure 8.9: Performance of confidence-thresholded MLLR-adapted models as a function of confidence threshold (rt06seval and rt07seval datasets).

testing reveals that these are significant improvements in the case of the word, phoneme and state-level confidence measures. The model-level confidence-weighted MLLR system gives a significant improvement over standard MLLR with 93.7% confidence, which is below the 95% confidence limit deemed as significant. No significant difference is found between any pair of confidence-weighted MLLR systems.

The improvements yielded by confidence-weighted MLLR over standard MLLR are smaller than those yielded by the optimally-thresholded systems of Section 8.7.1¹. However, when applying the MPSSWE significance test, no significant difference is found between any confidence-weighted MLLR system and, for example, the word-level confidence-thresholded MLLR system at a threshold of 0.9.

¹A similar result is found when comparing confidence-thresholded and confidence-weighted MLLR using the word-level maximal frame posterior confidence measure in Chapter 9 of Pitz (2005).

System		WER (%)		
		<i>rt06seval</i>	<i>rt07seval</i>	Average
Standard MLLR		31.4	36.2	33.9
Confidence measure	Word	31.3	36.1	33.8
	Phoneme	31.3	36.0	33.8
	Model	31.3	36.1	33.8
	State	31.3	36.1	33.8

Table 8.4: *Performance of confidence-weighted MLLR-adapted models (rt06seval and rt07seval datasets).*

8.7.3 Experiment 3: Ideal confidence-driven MLLR

To establish the optimal performance of confidence-driven MLLR with respect to each of the four different confidence measures, ideal confidence measures are used with the confidence-thresholded MLLR method². The ideal confidence measures are those which dismiss the frames of incorrectly transcribed data (with respect to the correct alignment) and retain the correctly-transcribed frames. So, for example, the ideal phoneme-level confidence measure dismisses all frames for which the phoneme label of the MAP alignment disagrees with the phoneme label of the correct alignment. The alignment of the correct transcription is used in this way to identify the ideal confidence measures at the word, phoneme, model and state levels.

The WER yielded after four iterations of confidence-thresholded MLLR using the ideal confidence measure at the word, phoneme, model and state levels is displayed in Table 8.5. All four ideal confidence-driven systems yield a significant performance improvement over

Confidence measure level	WER (%)		
	<i>rt06seval</i>	<i>rt07seval</i>	Average
Word	30.5	35.5	33.1
Phoneme	30.8	35.6	33.2
Model	30.6	35.4	33.1
State	30.5	35.4	33.1

Table 8.5: *Performance of ideal confidence-thresholded MLLR (rt07seval and rt07seval datasets).*

the optimal confidence-thresholded MLLR systems of Section 8.7.1 (an average WER of

²An ideal confidence measure can be defined as 1 if the label is correct and 0 otherwise. So confidence-weighted MLLR behaves identically to confidence-thresholded MLLR (with a threshold strictly between 0 and 1) under these ideal conditions.

33.7% in the case of all four confidence measures).

Moreover, the systems estimated using the ideal word, model and state-level measures yield a small but significant performance improvement over the system derived using the ideal phoneme-level measure. This result shows that use of ideal word, model and state-level confidence measures is preferable to use of the phoneme-level equivalent. Table 8.6 analyses the differences between the ideal phoneme-level and model-level confidence measures. In the case of the *rt06seval* and *rt07seval* datasets, the ideal phoneme-level confidence measure permits 7.99% of the frames rejected by the ideal model-level confidence measure. So when using the ideal phoneme-level confidence measure, the confidence-thresholded MLLR method increases the likelihood of incorrect models. This is an undesirable effect, and leads to the suboptimal performance of the ideal phoneme-level confidence-thresholded MLLR technique.

	frames (%)	
	retained at phoneme-level	dismissed at phoneme-level
retained at model-level	72.62	0.0
dismissed at model-level	7.99	19.38

Table 8.6: *Differences between ideal confidence measures at the phoneme and model levels (rt06seval and rt07seval datasets).*

No significant difference is found between the ideal state, model and word-level confidence-thresholded MLLR adapted systems. Comparing the frames of data retained and dismissed in each case, a high level of agreement is found between the ideal model and state-level confidence measures. As shown in Table 8.7, the measures agree for 97.76% of the frames in the *rt06seval* and *rt07seval* datasets. This high level of agreement causes the confidence-driven adaptation process to behave similarly and no significant performance difference between the resulting adapted systems is found. Less agreement (90.04% of all frames) is found

	frames (%)	
	retained at state-level	dismissed at state-level
retained at model-level	72.11	0.52
dismissed at model-level	1.73	25.65

Table 8.7: *Differences between ideal confidence measures at the state and model levels (rt06seval and rt07seval datasets).*

between the ideal word-level and model-level measures as shown in Table 8.8. Most of the disagreement (6.86% of all frames) is due to frames retained when using the ideal model-level confidence measure but dismissed when using the word-level confidence measure. A small volume (3.09% of all frames) of the frames which are dismissed using the ideal model-level

confidence measure are retained by the ideal word-level confidence measure. The inclusion of this small volume of frames which are dismissed using the ideal model-level confidence measure does not significantly degrade the performance of confidence-driven MLLR. Also, the exclusion of a reasonable (6.86%) volume of frames retained by the model-level confidence measure does not significantly affect the performance of confidence-driven MLLR.

	frames (%)	
	retained at word-level	dismissed at word-level
retained at model-level	65.76	6.86
dismissed at model-level	3.09	24.28

Table 8.8: *Differences between ideal confidence measures at the word and model levels (rt06seval and rt07seval datasets).*

8.7.4 Summary

The experimental results of this section show that word, phoneme, model and state-level confidence-driven MLLR adaptation deliver a small but significant performance improvement over standard unsupervised MLLR. Moreover, it has been shown that use of ideal word, model and state-level confidence measures yields significantly improved confidence-driven MLLR adaptation when compared to the ideal phoneme-level confidence measure. Although not the main focus of this chapter, the results of this section provide a useful comparison for the performance of confidence-driven MBRLR techniques in Section 8.8.

8.8 Evaluation: confidence-driven MBRLR

This section describes a series of experiments designed to evaluate aspects of confidence-driven unsupervised MBRLR adaptation. Standard MBRLR performance is measured in Section 8.8.1 to provide a performance baseline for confidence-driven MBRLR. Sections 8.8.2 and 8.8.3 evaluate the performance of word and sub-word confidence-thresholded MBRLR respectively. These results are compared with the performance of confidence-weighted MBRLR in Section 8.8.4. The optimal performance of the confidence-driven MBRLR technique is measured in Section 8.8.5 by using ideal confidence measures. Lastly, the impact of I-smoothing upon the performance of confidence-thresholded MBRLR is established in Section 8.8.6.

The multi-pass recognition system described in Section 8.5 is used, where the third-pass adaptation step is standard MBRLR or confidence-driven MBRLR. The MBRLR mean transforms are calculated as described in Section 5.3 and the symmetrically normalised frame error approximation (Section 6.3.1) to the Levenshtein error is used. Acoustic probability scaling, discussed in Section 5.2.5, is used within the MBRLR process. The acoustic

scale factor is $\frac{1}{14}$, the inverse of the language model scale factor used in the recognition pass. The learning rate constant E is set to 2 in all experiments. No I-smoothing is used unless specified otherwise. Preliminary experiments indicated that, with this configuration, twenty to thirty adaptation iterations are sufficient for convergence of the test set WER.

8.8.1 Experiment 1: Standard MBRLR

Before evaluating the confidence-driven MBRLR technique, the performance of standard unsupervised MBRLR adaptation is established. Table 8.9 compares the performance of unsupervised MLLR (four iterations) and MBRLR (twenty iterations) using the *rt06seval* and *rt07seval* test datasets. Note that three different MBRLR systems are evaluated, corresponding to the word, phoneme and state-level MBR criteria. The model-level MBR criterion is omitted due to its close similarity with the state-level MBR criterion (see Section 7.6). It is clear from Table 8.9 that the best performance is provided by MLLR adaptation,

System	WER (%)		
	<i>rt06seval</i>	<i>rt07seval</i>	Average
Unadapted	33.3	38.0	35.8
MLLR	31.4	36.2	33.9
MBRLR (word)	33.2	37.8	35.6
MBRLR (phoneme)	33.2	37.7	35.6
MBRLR (state)	33.2	37.7	35.6

Table 8.9: *Performance of standard MLLR and MBRLR (rt06seval and rt07seval datasets).*

which yields an average WER of 33.9%. This is a significant improvement over both the average unadapted system performance of 35.8% and the performance of all MBRLR-adapted systems. A smaller, but significant, performance improvement (35.6% average WER for all criteria) over the unadapted performance is provided by the MBRLR-adapted systems. No significant difference is found between the different MBRLR-adapted systems.

The superior performance of MLLR over MBRLR in the unsupervised scenario reveals an important issue, namely that the discriminative MBR adaptation technique is more sensitive to the quality of the unsupervised transcription than ML-based adaptation. This in turn is due to the multilateral action of the MBR technique; it is designed to increase the likelihood of model sequences of below-average error and to decrease the likelihood of all other model sequences. When the errors are incorrectly assigned, as is the case in unsupervised adaptation, the discriminative technique potentially increases the likelihood of model sequences of above-average error and potentially decreases the likelihood of model sequences of below-average error, including the correct model sequence. This undesirable effect is more constrained in the case of ML-based adaptation. In this case, when the reference model sequence is incorrectly assigned, the ML-based technique increases the likelihood of one of the incorrect model sequences but does not directly alter the likelihood

of the correct model sequence. Therefore MLLR adaptation displays greater robustness than MBRLR to errors in the estimated reference transcription, as the results of Table 8.9 confirm.

8.8.2 Experiment 2: Confidence-thresholded MBRLR

The experimental work described in this section is designed to compare the performance of confidence-thresholded MBRLR with standard MBRLR and to gain insight into the effect of the confidence threshold upon the adaptation process. The MBRLR formulation is fixed to the word-level MBR criterion and the confidence measure to the word-level maximal frame posterior. Figure 8.10 displays the WER yielded by the word-level confidence-thresholded MBRLR-adapted models after twenty adaptation iterations at a range of confidence thresholds between 0.1 and 0.99.

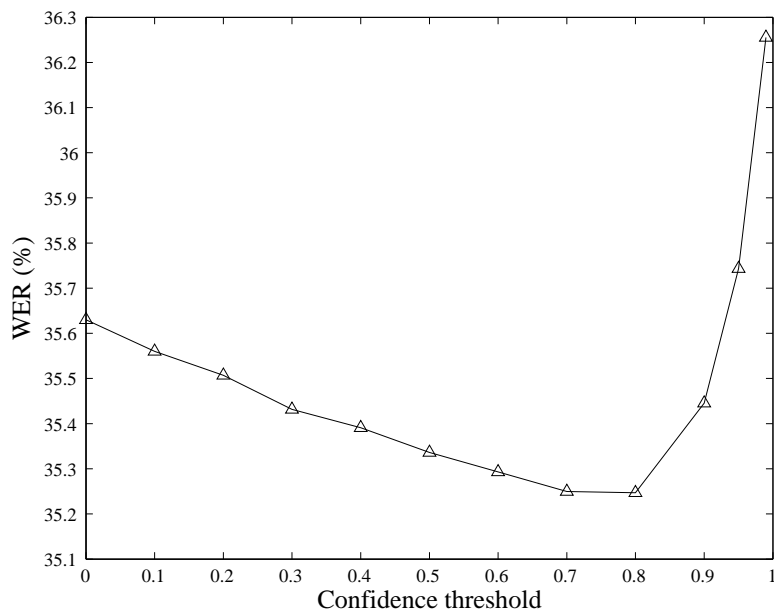


Figure 8.10: *Word-level confidence-thresholded MBRLR performance as a function of confidence threshold (rt06seval and rt07seval datasets).*

The threshold of 0.0 in Figure 8.10 corresponds to standard word-level MBRLR adaptation. As the threshold increases between the values of 0.0 and 0.8, a gradual decrease in WER (from 35.6% at a threshold of 0.0 to 35.2% at a threshold of 0.8) is observed. This demonstrates the benefit of ignoring errors associated with low-confidence labels of the reference transcription. As the confidence threshold increases from 0.8 to 0.99, a more

rapid increase in WER is witnessed. This illustrates the tradeoff between ignoring errors derived from low-confidence labels and ignoring all error information. An optimal word-level confidence threshold of approximately 0.8 exists for the dataset and system considered in this experiment. At this threshold a significant performance improvement over standard MBRLR is achieved.

8.8.3 Experiment 3: Sub-word confidence-thresholded MBRLR

In the experiment of Section 8.8.2 the confidence measure was fixed to the word-level maximal frame posterior. The impact of sub-word confidence measures is now compared with this word-level baseline. Note that the confidence measure and hypothesis space correspond, so, for example, a phoneme-level confidence measure is used in conjunction with the phoneme-level MBR criterion.

Figure 8.11 displays the WER yielded by the confidence-thresholded MBRLR-adapted models after twenty adaptation iterations at a range of confidence thresholds between 0.1 and 0.99. Each curve represents a different confidence-driven MBR criteria corresponding to the criterion and confidence measures at the word, phoneme and state levels. Note that the model-level criterion and confidence measure does not feature in these experiments due to the close similarity of both the state and model-level criteria (see Section 7.6) and the state and model-level confidence measures (see Section 8.6.2).

Each of the three curves achieves a minimum WER at a confidence threshold in the range of 0.8 to 0.9 inclusive. In all cases there is a tradeoff between ignoring errors derived from low-confidence labels of the reference transcription and ignoring all errors. Notice that, in the case of the phoneme-level formulation, the increase in WER at thresholds above 0.9 is less pronounced than the increase observed in the case of the word and state-level formulations. This is because a comparatively large volume of data has confidence above 0.9 in the case of the phoneme-level confidence measure, as displayed in Figure 8.7.

In the case of all three criteria considered, confidence-thresholded MBRLR at the optimal confidence threshold yields significantly improved performance over the performance of standard word-level MBRLR (35.6% WER); a WER of 35.4% in the case of phoneme-level MBRLR at a confidence threshold of 0.9 and a WER of 35.2% at the threshold of 0.8 in the case of both word and state-level MBRLR.

Further, the MPSSWE test reveals a significant difference between the state-level MBRLR formulation at the threshold of 0.8 and the phoneme-level formulation at 0.9. No other significant differences are found between the confidence-thresholded MBRLR-adapted systems at their optimal confidence thresholds. This significant difference between phoneme-level and state-level confidence-thresholded MBRLR is not obviously explicable. The superior performance of the state-level formulation cannot be attributed to any superiority of the state-level MBR criterion over the phoneme-level MBR criterion as no significant difference was observed between the performance of the state and phoneme-level MBR-estimated models in the experiments of Chapter 7. Nor can this difference be explained in terms of the superiority of state-level confidence measures since, as demonstrated in Section 8.6.1, no major difference was observed between the performance of the classifiers induced by the

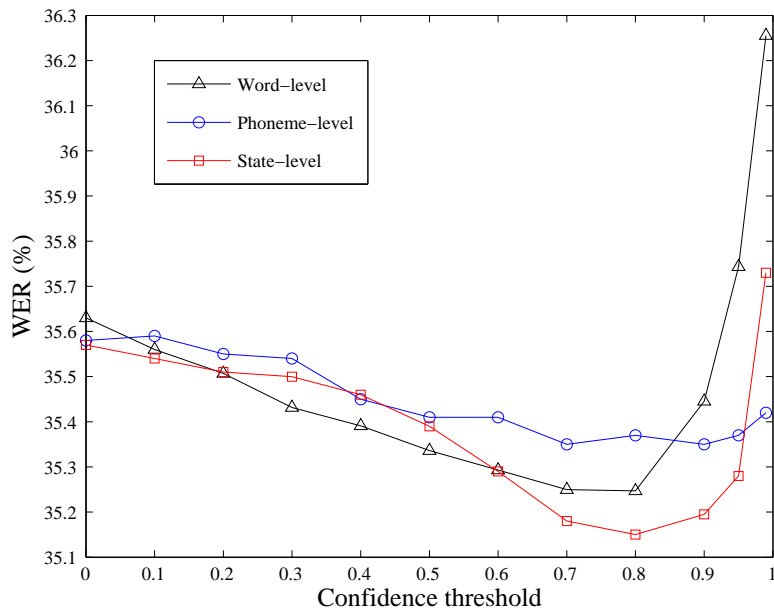


Figure 8.11: *Sub-word level confidence-thresholded MBRLR performance as a function of confidence threshold (rt06seval and rt07seval datasets).*

state and phoneme-level confidence measures.

Table 8.10 displays the differences between the frame-level classifiers induced by the state and phoneme-level maximal frame posterior confidence measures at thresholds of 0.8 and 0.95 respectively. Approximately the same volume of data is dismissed by each classifier (30.55% of all frames in the case of the state-level classifier and 28.64% in the case of the phoneme-level classifier). Additionally, very similar classification performance is yielded by these classifiers, as confirmed by the miss and false alarm rates displayed in Table 8.11.

Since the performance of state-level confidence-thresholded MBRLR at a threshold of 0.8 is significantly better than that of phoneme-level confidence-thresholded MBRLR at a threshold of 0.95, one is led to the following conclusion. The classifier induced by the state-level maximal frame posterior confidence measure (at a threshold of 0.8) makes errors (misses and false alarms) which are less harmful to the confidence-driven MBRLR adaptation process than the errors committed by the classifier induced by the phoneme-level maximal frame posterior confidence measure (at a threshold of 0.95). To understand why this is the case requires further research.

	frames (%)	
	retained at state-level	dismissed at state-level
retained at phoneme-level	62.52	8.83
dismissed at phoneme-level	6.92	21.72

Table 8.10: *Differences between classifiers induced by the state and phoneme-level maximal frame posterior confidence measures at thresholds of 0.8 and 0.95 respectively (rt06seval and rt07seval datasets).*

	frames (%)	
	Miss	False alarm
phoneme-level (0.95)	19.45	33.94
state-level (0.8)	19.20	37.41

Table 8.11: *Performance of classifiers induced by the state and phoneme-level maximal frame posterior confidence measures at thresholds of 0.8 and 0.95 respectively (rt06seval and rt07seval datasets).*

8.8.4 Experiment 4: Confidence-weighted MBRLR

Use of confidence-weighted MBRLR in the third-pass adaptation step of the evaluation system described in Section 8.5 yields the results presented in Table 8.12. Again, the MBR criterion and confidence measure correspond. So, for example, the system labelled ‘state-level’ deploys a confidence-weighted state-level MBR criterion, where the confidence measure used is the state-level maximal frame posterior. Twenty transform estimation iterations are performed in all cases.

System	WER (%)		
	rt06seval	rt07seval	Average
Word-level	33.1	37.6	35.4
Phoneme-level	33.2	37.7	35.5
State-level	33.2	37.6	35.5

Table 8.12: *Performance of confidence-weighted MBRLR (rt06seval and rt07seval datasets).*

Compare the performance of confidence-weighted MBRLR with the performance of standard unsupervised MBRLR, shown in Table 8.9. A small performance improvement is yielded over standard MBRLR (which yields 35.6% WER for all criterion formulations) in the case of all criteria and corresponding confidence measures. However confidence-weighted

MBRLR yields more modest WER gains compared to the optimal confidence-thresholded systems of Section 8.8.3.

Significance testing reveals significant differences between the performance of all three confidence-weighted MBRLR-adapted systems and the standard word-level MBRLR-adapted system. No significant difference is found between the performance of the three confidence-weighted MBRLR-adapted systems.

Significance testing also shows that, for example, the confidence-thresholded state-level MBRLR-adapted models, using a confidence threshold of 0.8, delivers significantly better performance than any of the confidence-weighted MBRLR-adapted models. Given the optimal confidence threshold, it is more beneficial to completely disregard, rather than to merely de-emphasise, errors derived from low-confidence reference labels.

8.8.5 Experiment 5: Ideal confidence-driven MBRLR

To establish the optimal performance of confidence-driven MBRLR adaptation techniques, an experiment using ideal confidence measures is conducted. The ideal confidence measure at the word, phoneme and state level is established by identifying incorrectly-transcribed segments of the aligned reference transcription, as described in Section 8.7.3.

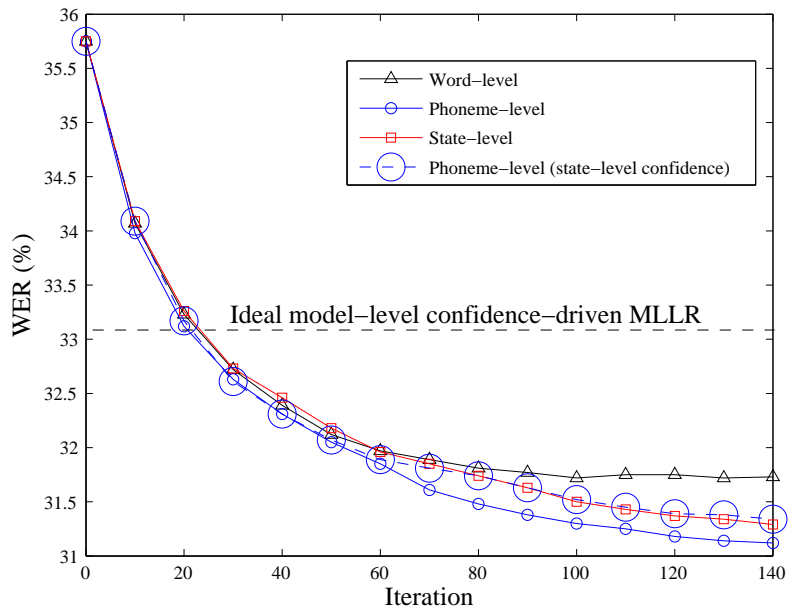


Figure 8.12: *Ideal confidence-thresholded MBRLR performance (rt06seval and rt07seval datasets).*

Figure 8.12 plots the WER of the adapted system when using ideal confidence-thresholded

MBRLR and the *rt06seval* and *rt07seval* test datasets. The horizontal axis represents the adaptation iteration. Each solid curve corresponds with a different MBR criterion formulation and the associated ideal confidence measure. For the purposes of comparison, the ideal model-level confidence-driven MLLR performance is also marked on the graph. This is the average WER of the ideal model-level confidence-driven system (33.1%), as shown in Table 8.5.

After thirty adaptation iterations, all ideal confidence-thresholded MBRLR formulations achieve a significantly lower WER than ideal confidence-driven MLLR. This is because the negative impact of the erroneous transcription has been constrained. Without this negative impact, the benefit of the discriminative MBRLR adaptation technique over the generative MLLR adaptation method becomes observable. Thus, given perfect knowledge of the correct and incorrectly-transcribed reference labels, confidence-driven MBRLR delivers performance significantly superior to that of confidence-driven MLLR.

Compare the performance of the ideal word, phoneme and state-level confidence-driven MBRLR adapted systems. After 140 iterations the WER of the word-level, state-level and phoneme-level systems converge to values of 31.7%, 31.3% and 31.1% respectively. Significance testing reveals significant differences between all three systems. The differences between these systems are due to two different factors; the relative generalisation of the different MBR criteria and the different volumes of data retained when using ideal word, phoneme, and state-level confidence measures.

As explained in Section 8.4.4, the issue of generalisation is relevant to the ideal confidence-thresholded unsupervised MBRLR adaptation task. Further, the experimental results of Chapter 7 have shown that significantly superior generalisation is provided by sub-word MBR criteria. As shown in Figure 8.12, sub-word MBR criteria again show superior generalisation to the word-level MBR criterion in the context of ideal confidence-thresholded unsupervised MBRLR adaptation.

However, no significant difference between the generalisation of phoneme-level and state-level MBR criteria was witnessed in the experimental results of Chapter 7. So the significant performance difference witnessed between ideal confidence-thresholded state and phoneme-level MBRLR adaptation cannot be attributed to differences in the criteria. Instead this difference must be attributed to a larger volume of data used in the case of the ideal phoneme-level confidence measure (80.61% of all frames, in comparison to 73.84% of all frames in the case of the ideal state-level confidence measure) as shown in Table 8.13.

	frames (%)	
	retained at phoneme-level	dismissed at phoneme-level
retained at state-level	73.73	0.11
dismissed at state-level	6.89	19.28

Table 8.13: *Differences between ideal confidence measures at the phoneme and state levels (rt06seval and rt07seval datasets).*

To verify this explanation, the performance of confidence-thresholded phoneme-level MBRLR is measured when using the ideal state-level confidence measure. The performance of this configuration is indicated by the dotted line labelled ‘phoneme-level (state-level confidence)’ in Figure 8.12. After 140 iterations, no significant difference is found between phoneme-level and state-level MBRLR, where both criteria use the ideal state-level confidence measure. This confirms that the phoneme and state-level MBR criteria display no significant difference in generalisation. It must therefore be concluded that the significant difference between ideal phoneme-level confidence-thresholded MBRLR and ideal state-level confidence-thresholded MBRLR is attributable to the use of different confidence measures.

Although some of the data (6.89% of all adaptation frames, see Table 8.13) retained by the ideal phoneme-level confidence measure corresponds to incorrectly-transcribed states (but the correct phoneme), useful phoneme-level discrimination is provided by these frames. This useful discrimination is absent from the ideal state-level confidence-thresholded system because these frames are dismissed from the adaptation process.

To summarise the results of this section, given an ideal confidence measure, confidence-thresholded MBRLR yields significant improvements over confidence-thresholded MLLR. Ideal sub-word level confidence-thresholded MBRLR delivers significant performance improvement over ideal word-level confidence-thresholded MBRLR due to the superior generalisation of sub-word MBR criteria. Lastly, ideal phoneme-level confidence-thresholded MBRLR delivers significant improvements over ideal state-level confidence-thresholded MBRLR due to the use of additional data excluded from the state-level formulation.

8.8.6 Experiment 6: Confidence-thresholded MBRLR and I-smoothing

The experiments described previously in this chapter have not used the I-smoothing technique for MBRLR, introduced in Section 5.3.1. An experimental evaluation of the impact of the I-smoothing method is now presented. As discussed in Section 8.8.5, the issue of generalisation is relevant to confidence-thresholded MBRLR. Since I-smoothing can improve the generalisation of MBR-estimated acoustic models, it is feasible that unsupervised confidence-thresholded MBRLR adaptation will also benefit from the use of I-smoothing.

Ideal confidence case

Figure 8.13 plots the performance of the first 140 iterations of I-smoothed confidence-thresholded phoneme-level MBRLR adaptation in the case of the ideal phoneme-level confidence threshold. Each curve corresponds to a different value of the constant τ (see Equation 5.3.6), as indicated in the legend.

The use of a zero-valued I-smoothing constant τ corresponds to no use of smoothing and the ideal phoneme-level confidence-thresholded performance of Figure 8.12. This curve converges at a WER of 31.1% after 140 iterations. As τ increases, the criterion approximates more closely the phoneme-level confidence-thresholded ML criterion and the transform re-estimation equations approximate more closely the phoneme-level confidence-thresholded MLLR transform re-estimation. So as τ increases, the associated curve approximates more

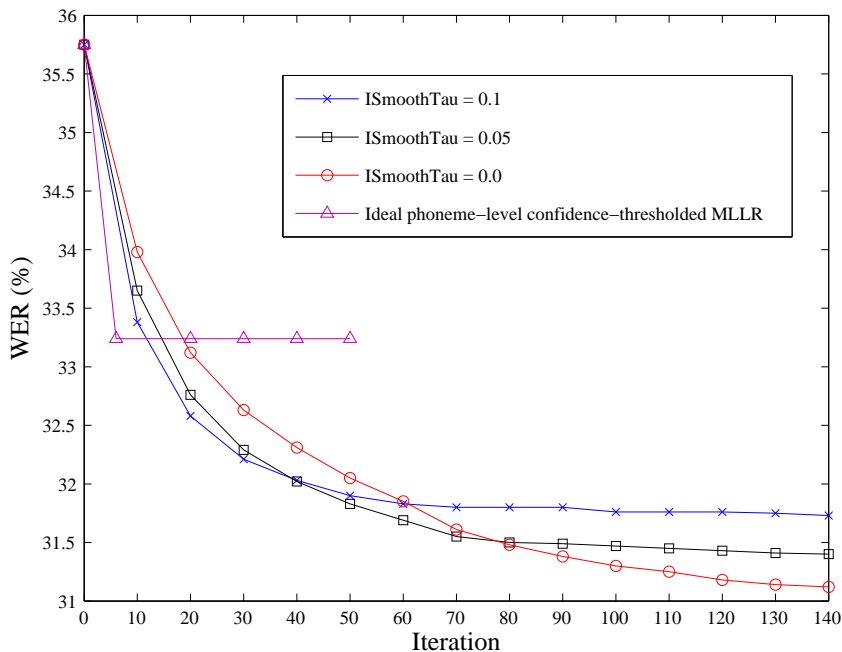


Figure 8.13: Performance of ideal phoneme-level confidence-thresholded MBRLR with I-smoothing (rt06seval and rt07seval datasets).

closely the curve corresponding to ideal phoneme-level confidence-driven MLLR. With sufficiently large τ , the discriminative part of the smoothed MBR criterion is effectively ignored and the transform re-estimation equations effectively implement MLLR transform re-estimation.

One side-effect of increasing τ is quicker convergence of the criterion. This is consistent with the relatively quick convergence of ML re-estimation procedures over discriminative re-estimation procedures. However no beneficial effect is observed in terms of performance for non-zero values of τ . At a τ value of 0.05 the WER converges to 31.4%, and at a τ value of 0.1 the WER converges to 31.7%. So, in the case of the ideal confidence measures, I-smoothed confidence-thresholded MBRLR fails to deliver improved performance over unsmoothed MBRLR.

Imperfect confidence case

Table 8.14 presents the performance yielded after twenty iterations of I-smoothed phoneme-level confidence-thresholded MBRLR for a range of smoothing factors τ . In this case the imperfect phoneme-level maximal frame posterior confidence measure at a threshold of 0.9

is used. Again, a τ value of 0.0 corresponds to no use of I-smoothing. An arbitrarily large value of τ corresponds to phoneme-level confidence-thresholded MLLR. As in the case of ideal confidence-thresholded MBRLR, as τ increases, the performance more closely approximates the performance of confidence-thresholded MLLR. In this case, increasing τ yields significant performance improvements. However, no value of τ delivers improvements over confidence-thresholded MLLR performance.

I-smooth constant τ	WER (%)		
	<i>rt06seval</i>	<i>rt07seval</i>	Average
0.0	33.0	37.5	35.4
0.05	32.3	36.9	34.7
0.1	32.2	36.6	34.5
1.0	31.3	36.1	33.8
∞ (confidence-thresholded MLLR)	31.2	36.0	33.7

Table 8.14: *Performance of I-smoothed phoneme-level confidence-thresholded MBRLR (threshold 0.9, rt06seval and rt07seval datasets).*

To summarise the results of this section, in the case of unsupervised MBRLR adaptation using the dataset considered in this experiment, and with an ideal confidence measure, I-smoothing delivers no improved generalisation over unsmoothed confidence-thresholded MBRLR adaptation. In the case of unsupervised MBRLR adaptation using the dataset considered in this experiment, and with an imperfect confidence measure, I-smoothed MBRLR delivers no improved generalisation over standard confidence-driven MLLR adaptation.

8.9 Summary and future work

This chapter has introduced and motivated novel confidence-driven MBR criteria by refining the approximate error function of the standard MBR criterion. Experimental evaluations on a large vocabulary recognition task have demonstrated that both confidence-thresholded and confidence-weighted unsupervised MBRLR adaptation deliver significant performance improvements over standard unsupervised MBRLR adaptation when using confidence measures based on frame posteriors.

While standard unsupervised MBRLR adaptation gives significantly inferior performance to standard unsupervised MLLR adaptation, it has been shown that, given an ideal confidence measure (as defined in Section 8.7.3), confidence-thresholded unsupervised MBRLR adaptation yields significantly superior performance to confidence-driven unsupervised MLLR.

Experimentation with confidence-thresholded MBRLR formulations corresponding to word and sub-word hypothesis spaces reveals that, in the case of ideal confidence measures, sub-word formulations yield significantly superior performance to ideal word-level

confidence-thresholded MBRLR. This is due to the superior generalisation of sub-word level MBR criteria over word-level MBR, as evidenced by the experimental results of Chapter 7. Moreover, ideal phoneme-level confidence-thresholded MBRLR yields significantly superior performance to ideal state-level confidence-thresholded MBRLR due to the use of additional data excluded by the state-level formulation.

When using confidence measures based on the frame posterior, state-level confidence-thresholded MBRLR shows significantly superior performance to phoneme-level confidence-thresholded MBRLR. Analysis of the different behaviour and performance of the state and phoneme-level maximal frame posterior confidence measures fails to account for this performance difference. Further analysis and experimentation is required to understand the impact of different errors made by the confidence-thresholded classifier upon the performance of the resulting confidence-thresholded MBRLR adaptation process.

Although the use of the I-smoothing technique is motivated with regard to unsupervised confidence-driven MBRLR adaptation, no performance improvement over unsmoothed MBRLR has been observed when using this technique in the case of ideal confidence measures. Nor has any improvement over confidence-thresholded MLLR adaptation been witnessed when using I-smoothed confidence-thresholded MBRLR adaptation with maximal frame posterior confidence measures.

8.9.1 Future work

Future exploratory work on MBRLR adaptation may investigate if techniques used to enhance the generalisation of standard MBR parameter estimation, e.g. acoustic scaling and choosing a less specific language model, are applicable also to supervised and unsupervised MBRLR. Additionally, the impact of different learning rates (the constant E in the implementation used in this thesis) upon MBRLR adaptation should be measured and understood.

There is much scope for future research in confidence-driven MBRLR adaptation. This work could focus on several different aspects of the adaptation technique. There is certainly a need to develop an understanding of the relationship between errors made by the confidence-based classifier (when classifying frames as correctly or incorrectly labelled) and the performance of the confidence-thresholded MBRLR adaptation process. For example, is a miss type of misclassification more harmful, in terms of performance, than a false alarm misclassification?

The field of confidence estimation is an area of research which directly impacts confidence-driven MBRLR. As evidenced by the experimental work of this chapter, with improved confidence measures, confidence-thresholded MBRLR adaptation has the capacity to yield performance which is superior to state-of-the-art confidence-driven MLLR adaptation. Future improvements on confidence measures are the key to access this superior performance.

Chapter 9

Conclusion

This thesis includes several contributions to the theory, implementation, understanding and performance of MBR acoustic model estimation and adaptation. In this chapter these contributions are summarised and questions which may be addressed by future research in this field are raised.

9.1 Contributions

In the first chapter the objectives of the thesis were stated. The contributions of the thesis are directly related to pursuit of these objectives and therefore these contributions are presented with respect to their corresponding objectives.

Objective 1: MBR criterion optimisation theory

The first objective of the thesis was to investigate if an auxiliary function can be specified which both justifies the MBR extended Baum-Welch parameter update equations and specifies the learning rate used in these equations. In Chapter 5 such an auxiliary function was presented. The proof that this function is an auxiliary function involves specification of a lower bound for the learning rate constant present in the MBR extended Baum-Welch parameter update equations.

Objective 2: MBR error approximation

The second thesis objective was to revisit the issue of error approximation with regard to the implementation of MBR acoustic model estimation. In Chapter 6, the symmetrically normalised frame error approximation was introduced to address some of the limitations of a previously used approximation. This approximation is shown to correlate with the Levenshtein error to a greater extent than the previously used approximation. Additionally, use of the novel approximation within MBR acoustic model estimation yields acoustic models which display significant classification performance improvements over models estimated using the previously used approximation.

Objective 3: Sub-word MBR criteria

The third objective of the thesis was to better understand the superior generalisation of phoneme-level MBR-estimated acoustic models over word-level MBR-estimated models. In Chapter 7, analysis of differences between the word and phoneme-level MBR criteria has led to motivation for use of the phoneme-level criterion. Further, analytical work has shown that the improvements yielded by the phoneme-level MBR criterion are at least partly due to more effective discrimination between speech and silence in the case of phoneme-level MBR-estimated models.

Further still, in the case of large vocabulary systems which deploy model-level parameter-tying, novel model and state-level formulations of the MBR criterion have been presented. Experimental work has shown that significant performance improvements over word-level MBR-estimated acoustic models are yielded by model and state-level MBR-estimated models.

Objective 4: Confidence-driven MBR acoustic model adaptation

The final thesis objective was to incorporate confidence information into the unsupervised MBR-based acoustic model adaptation technique and to quantify the performance of the resulting adapted models. In Chapter 8, confidence-driven MBR criteria have been introduced and applied to the task of unsupervised linear regression-based speaker adaptation. Significant performance improvements over the standard unsupervised MBRLR technique are witnessed when using confidence-driven MBRLR. Moreover, the experimental work illustrates that, with improved confidence measures, confidence-driven MBRLR adaptation has the capacity to deliver substantial performance improvements over the confidence-driven MLLR adaptation techniques which currently provide state-of-the-art unsupervised adaptation performance.

9.2 Future work

In Sections 9.2.1, 9.2.2, 9.2.3 and 9.2.4, future research related directly to the content of this thesis is discussed. More general future work related to discriminative criteria is discussed in Section 9.2.5.

9.2.1 Auxiliary function for MBR linear regression

The theoretical work of this thesis has identified an auxiliary function for the MBR criterion with respect to the parameters of Gaussian state output distributions of HMMs. However there is no guarantee that this auxiliary function is applicable to the parameters of the affine transforms used in MBR linear regression acoustic model adaptation. Future theoretical work may identify an auxiliary function for MBR linear regression transforms.

9.2.2 Error metrics and approximations

The experimental evidence of Chapter 6 has shown that significant improvements in the generalisation of MBR-estimated acoustic models may be achieved via use of a more accurate approximation to the Levenshtein error. However, it is unknown if the Levenshtein error metric itself is an optimal error function, with respect to the generalisation of the MBR criterion deploying this error metric. Indeed, some recent research in MBR-based acoustic model estimation (Zheng and Stolcke (2005), Du et al. (2006)) has introduced alternative error functions and reported improvements in the generalisation of the estimated acoustic models. However, this research has used the suboptimal baseline approximate error (as defined in Chapter 6) as a performance baseline. It may prove informative to compare the performance of the techniques described in Zheng and Stolcke (2005) and Du et al. (2006) with the performance of MBR models estimated using the symmetrically normalised frame error approximation presented in this thesis. Further, as mentioned in Chapter 6, a comparison between alignment-based error approximations and the lattice segmentation approach to error approximation may prove instructive.

9.2.3 Sub-word MBR criteria

The arguments presented in Chapter 7 motivate the use of model and state-level formulations of the MBR criterion for acoustic models which deploy parameter-tying at the model and state levels. Using large vocabulary acoustic models with highly constrained parameter-tying, a high level of correlation is found between errors at these detailed levels and phoneme-level errors. Consequently, similar acoustic models are yielded by phoneme, model and state-level MBR estimation. Future research should compare the acoustic models yielded by these different MBR formulations when fewer constraints are placed upon the acoustic parameter-tying scheme.

9.2.4 Confidence-driven MBRLR

The MBRLR adaptation method is a recently proposed technique and consequently there is much scope for future research on this topic. In the case of unsupervised MBRLR adaptation, where the adaptation and test data coincide, questions arise with regard to the generalisation of the method. Since the adaptation and test data coincide, overfitting the adaptation data does not necessarily lead to poor test set performance. However, as explained in Chapter 8, the issue of generalisation is relevant to confidence-driven MBRLR. Future research may investigate if techniques used to enhance the generalisation of standard MBR parameter estimation, for example acoustic scaling and choosing a less specific language model, can also be applied to confidence-driven MBRLR.

As demonstrated by the experimental evidence of Chapter 8, future research yielding improved confidence estimation in ASR has direct implications for the performance of confidence-driven MBRLR adaptation. More subtly, the relationship between errors committed by confidence estimation methods and the performance of confidence-driven MBRLR adaptation is at the moment unknown, and hence a possible direction of future research.

9.2.5 Future discriminative criteria

In the field of ASR, the group of successful discriminative estimation criteria may be seen as a disparate set. However in Macherey et al. (2005), a general discriminative criterion is presented which subsumes the conditional ML, MCE, and MBR criteria. Additionally, there are well-understood links between MCE and large margin estimation (Yu et al. (2008)).

While such connections exist between discriminative criteria, relatively little research has been conducted into criteria combination. For example, a combination of the MBR and MCE criteria, which introduces knowledge of the word or phoneme-level errors associated with competing hypotheses into the MCE criterion, is theoretically well-motivated. This is because such a criterion contains appealing properties of MCE which are absent from MBR (namely that discrimination between hypotheses which are close to decision boundaries is explicitly emphasised) as well as properties of MBR which are absent from MCE (namely that the reduction of the posterior of hypotheses with large error is explicitly emphasised). While such a formulation has been proposed beforehand (McDermott (1997)), to the author's knowledge no implementation and evaluation of such a criterion exists to date.

Appendix A

MBR criterion optimisation

In this appendix, arguments similar to those used in Gunawardana (2001) and Axelrod et al. (2007) on the auxiliary function for the conditional ML criterion are presented to prove the existence of an auxiliary function for the Bayes risk criterion with respect to the mean and covariance parameters of a continuous density HMM. Then, using this auxiliary function, the extended Baum-Welch re-estimation equation for the mean update of a continuous density HMM is derived.

Minimum Bayes risk (MBR) acoustic parameter estimation attempts to find acoustic model parameters θ_{MBR} satisfying Equation A.0.1.

$$\begin{aligned}
 \theta_{\text{MBR}} &= \arg \min_{\theta} R_{\text{MBR}}(\theta) \\
 &= \arg \min_{\theta} \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} p(w_1^N | \hat{\mathbf{o}}_1^{T(r)}, \theta) L(w_1^N, \hat{w}_1^{M(r)}) \\
 &= \arg \min_{\theta} \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} \frac{p(\hat{\mathbf{o}}_1^{T(r)} | w_1^N, \theta) p(w_1^N) L(w_1^N, \hat{w}_1^{M(r)})}{p(\hat{\mathbf{o}}_1^{T(r)} | \theta)} \quad (\text{A.0.1})
 \end{aligned}$$

Here $\hat{\mathbf{o}}_1^{T(r)}$ is the r -th training set observation sequence, $\hat{w}_1^{M(r)}$ is the corresponding transcription, and θ represents the acoustic model parameters. \mathcal{W} is the set of all possible word sequences. The MBR criterion of Equation A.0.1 may be rewritten as in Equation A.0.2.

$$R_{\text{MBR}}(\theta) = \frac{\sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} p(\hat{\mathbf{o}}_1^{T(r)} | w_1^N, \theta) p(w_1^N) L(w_1^N, \hat{w}_1^{M(r)}) \prod_{k=1; k \neq r}^R p(\hat{\mathbf{o}}_1^{T(k)} | \theta)}{\prod_{k=1}^R p(\hat{\mathbf{o}}_1^{T(k)} | \theta)} \quad (\text{A.0.2})$$

A.1 Preliminary theorems

To derive a re-estimation equation for the parameters θ_{MBR} of Equation A.0.1 the following lemma, derived in Gunawardana (2001), is used¹.

¹Note that the lemma here states a stronger two-way implication than the one-way implication proven in Gunawardana (2001). The two-way implication can be proven in a similar way.

Lemma 1. Let $P(\theta) = \frac{Q(\theta)}{R(\theta)}$ be the ratio of two positive real-valued functions on the set Θ . Let $\theta' \in \Theta$ and define $G(\theta|\theta') = Q(\theta) - P(\theta')R(\theta) + D$ for any real constant D . Then the following equivalence holds.

$$G(\theta|\theta') \geq G(\theta'|\theta') \iff P(\theta) \geq P(\theta').$$

The following lemma follows immediately from Lemma 1.

Lemma 2. Let θ and θ' be elements of the parameter space describing the means and variances of the mixture components in a continuous density HMM. Define $Q(\theta)$ and $R(\theta)$ as the numerator and denominator of $R_{\text{MBR}}(\theta)$ respectively.

$$Q(\theta) = \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} p(\hat{\mathbf{o}}_1^{T(r)}|w_1^N, \theta) p(w_1^N) L(w_1^N, \hat{w}_1^{M(r)}) \prod_{k=1; k \neq r}^R p(\hat{\mathbf{o}}_1^{T(k)}|\theta) \quad (\text{A.1.1})$$

$$R(\theta) = \prod_{k=1}^R p(\hat{\mathbf{o}}_1^{T(k)}|\theta) \quad (\text{A.1.2})$$

Defining $P(\theta') = \frac{Q(\theta')}{R(\theta')}$ and $G(\theta|\theta') = Q(\theta) - P(\theta')R(\theta) + D$ in accordance with Lemma 1 gives Equation A.1.3.

$$\begin{aligned} G(\theta|\theta') &= \sum_{r=1}^R \prod_{k=1; k \neq r}^R p(\hat{\mathbf{o}}_1^{T(k)}|\theta) \sum_{w_1^N \in \mathcal{W}} p(\hat{\mathbf{o}}_1^{T(r)}|w_1^N, \theta) p(w_1^N) L(w_1^N, \hat{w}_1^{M(r)}) \\ &\quad - \sum_{r=1}^R \prod_{k=1}^R p(\hat{\mathbf{o}}_1^{T(k)}|\theta) \sum_{w_1^N \in \mathcal{W}} \frac{p(\hat{\mathbf{o}}_1^{T(k)}|w_1^N, \theta') p(w_1^N) L(w_1^N, \hat{w}_1^{M(r)})}{p(\hat{\mathbf{o}}_1^{T(r)}|\theta')} + D \\ &= \sum_{r=1}^R \prod_{k=1; k \neq r}^R p(\hat{\mathbf{o}}_1^{T(k)}|\theta) \sum_{w_1^N \in \mathcal{W}} L(w_1^N, \hat{w}_1^{M(r)}) \left[p(\hat{\mathbf{o}}_1^{T(r)}|w_1^N, \theta) p(w_1^N) \right. \\ &\quad \left. - p(w_1^N|\hat{\mathbf{o}}_1^{T(r)}, \theta') p(\hat{\mathbf{o}}_1^{T(r)}|\theta) \right] + D \end{aligned} \quad (\text{A.1.3})$$

Using Lemma 1, the following equivalence holds.

$$G(\theta'|\theta') - G(\theta|\theta') \geq 0 \iff R_{\text{MBR}}(\theta') \geq R_{\text{MBR}}(\theta) \quad (\text{A.1.4})$$

A.2 Definition of MBR auxiliary function

Let θ and θ' be elements of the parameter space describing the means and covariances of the mixture components in a continuous density HMM. Let \mathcal{S} represent the set of possible hidden Markov model state sequences S associated with the training observation sequences. Note that \mathcal{S} is the Cartesian product of state sequence spaces $\mathcal{S}_1 \times \mathcal{S}_2 \times \dots \times \mathcal{S}_r \times \dots \times \mathcal{S}_R$ where \mathcal{S}_r is the set of state sequences of length equal to the length of the r -th training

observation sequence, $T(r)$. Here a state sequence s_1^T is the catenation of R sequences, $s_1^{T(1)} s_1^{T(2)} \dots s_1^{T(r)} \dots s_1^{T(R)}$, where the index corresponds to the training example index.

The function $F_{\text{MBR}}(\theta, \theta', D)$ is defined in Equation A.2.1.

$$F_{\text{MBR}}(\theta, \theta', D) = \int_{\mathbf{o}_1^T} \sum_{s_1^T \in \mathcal{S}} f(\mathbf{o}_1^T, s_1^T, \theta' | \theta') \log f(\mathbf{o}_1^T, s_1^T, \theta | \theta') d\mathbf{o}_1^T \quad (\text{A.2.1})$$

Here $f(\mathbf{o}_1^T, s_1^T, \theta | \theta')$ is defined by Equation A.2.2 and D is defined by Equation A.2.3.

$$f(\mathbf{o}_1^T, s_1^T, \theta | \theta') = p(\mathbf{o}_1^T, s_1^T | \theta) \left[\sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} \mathbf{1}_{\hat{\mathbf{o}}_1^T}(\mathbf{o}_1^T) a(w_1^N, \hat{w}_1^{M(r)}, \hat{\mathbf{o}}_1^{T(r)}, s_1^{T(r)}, \theta') + d(s_1^T) \right] \quad (\text{A.2.2})$$

$$D = \int_{\mathbf{o}_1^T} \sum_{s_1^T \in \mathcal{S}} p(\mathbf{o}_1^T, s_1^T | \theta) d(s_1^T) d\mathbf{o}_1^T = \sum_{s_1^T \in \mathcal{S}} p(s_1^T) d(s_1^T) \quad (\text{A.2.3})$$

Above $\mathbf{1}_{\hat{\mathbf{o}}_1^T}(\mathbf{o}_1^T)$ is an indicator function and $\hat{\mathbf{o}}_1^T$ is the catenation of the observation sequences corresponding the training utterances, $\hat{\mathbf{o}}_1^{T(1)} \hat{\mathbf{o}}_1^{T(2)} \dots \hat{\mathbf{o}}_1^{T(R)}$.

The quantity $a(w_1^N, \hat{w}_1^{M(r)}, \hat{\mathbf{o}}_1^{T(r)}, s_1^{T(r)}, \theta')$ is defined in Equation A.2.4.

$$a(w_1^N, \hat{w}_1^{M(r)}, \hat{\mathbf{o}}_1^{T(r)}, s_1^{T(r)}, \theta') = L(w_1^N, \hat{w}_1^{M(r)}) \left[p(w_1^N | \hat{\mathbf{o}}_1^{T(r)}, \theta') - p(w_1^N | s_1^{T(r)}) \right] \quad (\text{A.2.4})$$

It is intended to prove that, given a large enough value for the constants $d(s_1^T)$, the function $F_{\text{MBR}}(\theta, \theta', D)$ is a valid auxiliary function for the MBR criterion $R_{\text{MBR}}(\theta)$. This is proved using an argument similar to the one presented in Axelrod et al. (2007) for the case of the conditional ML criterion. The function $F_{\text{MBR}}(\theta, \theta', D)$ is firstly manipulated into a form which is more useful for the purposes of this proof.

A.2.1 Manipulation of MBR auxiliary function

The quantity $\log p(\mathbf{o}_t | s_t, \theta')$ may be expressed as in Equation A.2.5 where $\boldsymbol{\mu}_{s_t}$ is the mean of state s_t , \mathbf{C}_{s_t} is the covariance matrix of state s_t and d is the dimension of the acoustic feature space.

$$\log p(\mathbf{o}_t | s_t, \theta') = \frac{1}{2} [\log \det(\mathbf{C}_{s_t}^{-1}) - d \log(2\pi) - (\mathbf{o}_t - \boldsymbol{\mu}_{s_t})^\top \mathbf{C}_{s_t}^{-1} (\mathbf{o}_t - \boldsymbol{\mu}_{s_t})] \quad (\text{A.2.5})$$

It can be shown that the quantity $\log p(\mathbf{o}_t | s_t, \theta')$ may also be expressed as in Equation A.2.6.

$$\log p(\mathbf{o}_t | s_t, \theta') = \boldsymbol{\lambda}_{s_t}^\top \mathbf{f}(\mathbf{o}_t) + K(\boldsymbol{\lambda}_{s_t}) \quad (\text{A.2.6})$$

Here $\boldsymbol{\lambda}_{s_t}$, $\mathbf{f}(\mathbf{o}_t)$ and $K(\boldsymbol{\lambda}_{s_t})$ are defined in equations A.2.7, A.2.8 and A.2.9 respectively.

$$\boldsymbol{\lambda}_{s_t} = \begin{bmatrix} \text{vec}(\mathbf{C}_{s_t}^{-1}) \\ \mathbf{C}_{s_t}^{-1} \boldsymbol{\mu}_{s_t} \end{bmatrix} \quad (\text{A.2.7})$$

$$\mathbf{f}(\mathbf{o}_t) = \begin{bmatrix} -\frac{1}{2}\text{vec}(\mathbf{o}_t\mathbf{o}_t^\top) \\ \mathbf{o}_t \end{bmatrix} \quad (\text{A.2.8})$$

$$K(\boldsymbol{\lambda}_{s_t}) = \frac{1}{2} \left[-d \log(2\pi) + \log \det(\mathbf{C}_{s_t}^{-1}) - (\mathbf{C}_{s_t}^{-1} \boldsymbol{\mu}_{s_t})^\top \mathbf{C}_{s_t} (\mathbf{C}_{s_t}^{-1} \boldsymbol{\mu}_{s_t}) \right] \quad (\text{A.2.9})$$

In the above equations, the notation $\text{vec}(\mathbf{X})$ is relevant to a symmetric $d \times d$ matrix \mathbf{X} . The symbol $\text{vec}(\mathbf{X})$ represents a column vector whose elements are the $d(d+1)/2$ upper triangular elements of the matrix \mathbf{X} written in some fixed order, and with the off-diagonal elements multiplied by $\sqrt{2}$. With this definition, the inner product $\text{vec}(\mathbf{X}_1)^\top \text{vec}(\mathbf{X}_2)$ (where \mathbf{X}_1 and \mathbf{X}_2 are symmetric) is equal to $\text{tr}(\mathbf{X}_1 \mathbf{X}_2)$, where $\text{tr}(\mathbf{X})$ represents the trace of matrix \mathbf{X} .

Using the definition of $F_{\text{MBR}}(\theta, \theta', D)$, the probability $p(\mathbf{o}_1^T, s_1^T | \theta)$ is re-expressed as the product $p(\mathbf{o}_1^T | s_1^T, \theta) p(s_1^T)$. Note that the fact that the parameters θ do not govern the HMM state transition probabilities has been used. Then, expanding this logarithm of the product term into a sum of logarithms, $F_{\text{MBR}}(\theta, \theta', D)$ may be expressed as in Equation A.2.10, where $H(\theta, \theta')$ and $I(\theta')$ are defined by Equations A.2.11 and A.2.12 respectively. Note that $I(\theta')$ is a function only of the parameters θ' , while $H(\theta, \theta')$ is a function of both θ and θ' .

$$F_{\text{MBR}}(\theta, \theta', D) = H(\theta, \theta') + I(\theta') \quad (\text{A.2.10})$$

$$H(\theta, \theta') = \int_{\mathbf{o}_1^T} \sum_{s_1^T \in \mathcal{S}} p(\mathbf{o}_1^T, s_1^T | \theta') \left[\sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} 1_{\hat{\mathbf{o}}_1^T(\mathbf{o}_1^T)} a(w_1^N, \hat{w}_1^{M(r)}, \hat{\mathbf{o}}_1^{T(r)}, s_1^{T(r)}, \theta') + d(s_1^T) \right] \log p(\mathbf{o}_1^T | s_1^T, \theta) d\mathbf{o}_1^T \quad (\text{A.2.11})$$

$$I(\theta') = \int_{\mathbf{o}_1^T} \sum_{s_1^T \in \mathcal{S}} p(\mathbf{o}_1^T, s_1^T | \theta') \left[\sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} 1_{\hat{\mathbf{o}}_1^T(\mathbf{o}_1^T)} a(w_1^N, \hat{w}_1^{M(r)}, \hat{\mathbf{o}}_1^{T(r)}, s_1^{T(r)}, \theta') + d(s_1^T) \right] \log \left[p(s_1^T) \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} \left[1_{\hat{\mathbf{o}}_1^T(\mathbf{o}_1^T)} a(w_1^N, \hat{w}_1^{M(r)}, \hat{\mathbf{o}}_1^{T(r)}, s_1^{T(r)}, \theta') + d(s_1^T) \right] \right] d\mathbf{o}_1^T \quad (\text{A.2.12})$$

These definitions are now used to express $H(\theta, \theta')$ as shown in Equation A.2.13, where the outer sum is over all states s .

$$H(\theta, \theta') = \sum_s \left[\gamma_s K(\boldsymbol{\lambda}_s) + \boldsymbol{\lambda}_s^\top \boldsymbol{\Gamma}_s \right] \quad (\text{A.2.13})$$

In the above equation, γ_s and $\mathbf{\Gamma}_s$ are as defined in Equations A.2.14 and A.2.15 respectively.

$$\gamma_s = p(\hat{\mathbf{o}}_1^T | \theta') \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} L(w_1^N, \hat{w}_1^{M(r)}) p(w_1^N | \hat{\mathbf{o}}_1^{T(r)}, \theta') [\gamma_s(r) - \gamma_s(r, w_1^N)] + D_s \quad (\text{A.2.14})$$

$$\mathbf{\Gamma}_s = p(\hat{\mathbf{o}}_1^T | \theta') \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} L(w_1^N, \hat{w}_1^{M(r)}) p(w_1^N | \hat{\mathbf{o}}_1^{T(r)}, \theta') [\mathbf{\Gamma}_s(r) - \mathbf{\Gamma}_s(r, w_1^N)] + D_s E_{\theta'_s} \{\mathbf{f}(\mathbf{o}_1^T)\} \quad (\text{A.2.15})$$

The quantity D_s is defined in Equation A.2.16 and the expected value $E_{\theta'_s} \{\mathbf{f}(\mathbf{o}_1^T)\}$ is defined in Equation A.2.17, where θ'_s represents the parameters of θ' associated with state s .

$$D_s = \sum_{t=1}^T \sum_{s_1^T \in \mathcal{S}} p(s_1^T) d(s_1^T) 1_s(s_t) \quad (\text{A.2.16})$$

$$E_{\theta'_s} \{\mathbf{f}(\mathbf{o}_1^T)\} = \int_{\mathbf{o}_1^T} p(\mathbf{o}_1^T | s, \theta'_s) \mathbf{f}(\mathbf{o}_1^T) d\mathbf{o}_1^T \quad (\text{A.2.17})$$

The above quantities $\gamma_s(r)$, $\gamma_s(r, w_1^N)$, $\mathbf{\Gamma}_s(r)$ and $\mathbf{\Gamma}_s(r, w_1^N)$ are defined in Equations A.2.18, A.2.19, A.2.20 and A.2.21 respectively.

$$\gamma_s(r) = \sum_{s_1^{T(r)} \in \mathcal{S}_r} p(s_1^{T(r)} | \hat{\mathbf{o}}_1^{T(r)}, \theta') \sum_{t=1}^{T(r)} 1_s(s_t(r)) \quad (\text{A.2.18})$$

$$\gamma_s(r, w_1^N) = \sum_{s_1^{T(r)} \in \mathcal{S}_r} p(s_1^{T(r)} | w_1^N, \hat{\mathbf{o}}_1^{T(r)}, \theta') \sum_{t=1}^{T(r)} 1_s(s_t(r)) \quad (\text{A.2.19})$$

$$\mathbf{\Gamma}_s(r) = \sum_{s_1^{T(r)} \in \mathcal{S}_r} p(s_1^{T(r)} | \hat{\mathbf{o}}_1^{T(r)}, \theta') \sum_{t=1}^{T(r)} 1_s(s_t(r)) \mathbf{f}(\mathbf{o}_t(r)) \quad (\text{A.2.20})$$

$$\mathbf{\Gamma}_s(r, w_1^N) = \sum_{s_1^{T(r)} \in \mathcal{S}_r} p(s_1^{T(r)} | w_1^N, \hat{\mathbf{o}}_1^{T(r)}, \theta') \sum_{t=1}^{T(r)} 1_s(s_t(r)) \mathbf{f}(\mathbf{o}_t(r)) \quad (\text{A.2.21})$$

Equation A.2.13 is derived in the following section.

Validity of Equation A.2.13

To demonstrate the validity of Equation A.2.13, the definition of $H(\theta, \theta')$ (Equation A.2.11) is firstly expanded in Equation A.2.22.

$$\begin{aligned}
& H(\theta, \theta') \\
&= \int_{\mathbf{o}_1^T} \left[\sum_{s_1^T \in \mathcal{S}} p(\mathbf{o}_1^T, s_1^T | \theta') \left[\sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} 1_{\hat{\mathbf{o}}_1^T}(\mathbf{o}_1^T) a(w_1^N, \hat{w}_1^{M(r)}, \hat{\mathbf{o}}_1^{T(r)}, s_1^{T(r)}, \theta') \right. \right. \\
&\quad \left. \left. + d(s_1^T) \right] \log [p(\mathbf{o}_1^T | s_1^T, \theta)] \right] d\mathbf{o}_1^T \\
&= \int_{\mathbf{o}_1^T} \left[\sum_{s_1^T \in \mathcal{S}} p(\mathbf{o}_1^T, s_1^T | \theta') \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} 1_{\hat{\mathbf{o}}_1^T}(\mathbf{o}_1^T) a(w_1^N, \hat{w}_1^{M(r)}, \hat{\mathbf{o}}_1^{T(r)}, s_1^{T(r)}, \theta') \log p(\mathbf{o}_1^T | s_1^T, \theta) \right. \\
&\quad \left. + \sum_{s_1^T \in \mathcal{S}} p(\mathbf{o}_1^T, s_1^T | \theta') d(s_1^T) \log p(\mathbf{o}_1^T | s_1^T, \theta) \right] d\mathbf{o}_1^T \tag{A.2.22}
\end{aligned}$$

The terms $X(\theta, \theta')$ and $Y(\theta, \theta')$ are then defined as follows, such that $H(\theta, \theta')$ is the sum of $X(\theta, \theta')$ and $Y(\theta, \theta')$.

$$\begin{aligned}
& X(\theta, \theta') \\
&= \int_{\mathbf{o}_1^T} \sum_{s_1^T \in \mathcal{S}} p(\mathbf{o}_1^T, s_1^T | \theta') \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} 1_{\hat{\mathbf{o}}_1^T}(\mathbf{o}_1^T) a(w_1^N, \hat{w}_1^{M(r)}, \hat{\mathbf{o}}_1^{T(r)}, s_1^{T(r)}, \theta') \log p(\mathbf{o}_1^T | s_1^T, \theta) d\mathbf{o}_1^T \tag{A.2.23}
\end{aligned}$$

$$Y(\theta, \theta') = \int_{\mathbf{o}_1^T} \sum_{s_1^T \in \mathcal{S}} p(\mathbf{o}_1^T, s_1^T | \theta') d(s_1^T) \log p(\mathbf{o}_1^T | s_1^T, \theta) d\mathbf{o}_1^T \tag{A.2.24}$$

By definition of the Dirac delta function, $X(\theta, \theta')$ may be re-expressed as shown by Equation A.2.25.

$$\begin{aligned}
& X(\theta, \theta') \\
&= \sum_{s_1^T \in \mathcal{S}} p(\hat{\mathbf{o}}_1^T, s_1^T | \theta') \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} a(w_1^N, \hat{w}_1^{M(r)}, \hat{\mathbf{o}}_1^{T(r)}, s_1^{T(r)}, \theta') \log p(\hat{\mathbf{o}}_1^T | s_1^T, \theta) \\
&= p(\hat{\mathbf{o}}_1^T | \theta') \sum_{s_1^T \in \mathcal{S}} p(s_1^T | \hat{\mathbf{o}}_1^T, \theta') \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} a(w_1^N, \hat{w}_1^{M(r)}, \hat{\mathbf{o}}_1^{T(r)}, s_1^{T(r)}, \theta') \log p(\hat{\mathbf{o}}_1^T | s_1^T, \theta) \tag{A.2.25}
\end{aligned}$$

Expanding Equation A.2.25 over the set of state sequences S_k corresponding to the individual utterances (indexed by k) gives Equation A.2.26.

$$\begin{aligned} & \frac{X(\theta, \theta')}{p(\hat{\mathbf{o}}_1^T | \theta')} \\ &= \prod_{k=1}^R \sum_{s_1^{T(k)} \in \mathcal{S}_k} p(s_1^{T(k)} | \hat{\mathbf{o}}_1^{T(k)}, \theta') \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} a(w_1^N, \hat{w}_1^{M(r)}, \hat{\mathbf{o}}_1^{T(r)}, s_1^{T(r)}, \theta') \log p(\hat{\mathbf{o}}_1^T | s_1^T, \theta) \end{aligned} \quad (\text{A.2.26})$$

Substituting Equation A.2.6 in Equation A.2.26 gives Equation A.2.27, where T is the total number of observations in sequence $\hat{\mathbf{o}}_1^T$, $\hat{\mathbf{o}}_t$ is the t -th observation in sequence $\hat{\mathbf{o}}_1^T$, and s_t is the t -th state in state sequence s_1^T .

$$\begin{aligned} & \frac{X(\theta, \theta')}{p(\hat{\mathbf{o}}_1^T | \theta')} \\ &= \prod_{k=1}^R \sum_{s_1^{T(k)} \in \mathcal{S}_k} p(s_1^{T(k)} | \hat{\mathbf{o}}_1^{T(k)}, \theta') \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} a(w_1^N, \hat{w}_1^{M(r)}, \hat{\mathbf{o}}_1^{T(r)}, s_1^{T(r)}, \theta') \sum_{t=1}^T \log p(\hat{\mathbf{o}}_t | s_t, \theta) \\ &= \prod_{k=1}^R \sum_{s_1^{T(k)} \in \mathcal{S}_k} p(s_1^{T(k)} | \hat{\mathbf{o}}_1^{T(k)}, \theta') \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} a(w_1^N, \hat{w}_1^{M(r)}, \hat{\mathbf{o}}_1^{T(r)}, s_1^{T(r)}, \theta') \sum_{t=1}^T [\boldsymbol{\lambda}_{s_t}^\top \mathbf{f}(\hat{\mathbf{o}}_t) \\ & \quad + K(\boldsymbol{\lambda}_{s_t})] \end{aligned} \quad (\text{A.2.27})$$

To further simplify the terms of Equation A.2.27, define $X_1(\theta, \theta')$ and $X_2(\theta, \theta')$ such that $X(\theta, \theta')$ is equal to $X_1(\theta, \theta') - X_2(\theta, \theta')$.

$$\begin{aligned} & \frac{X_1(\theta, \theta')}{p(\hat{\mathbf{o}}_1^T | \theta')} \\ &= \prod_{k=1}^R \sum_{s_1^{T(k)} \in \mathcal{S}_k} p(s_1^{T(k)} | \hat{\mathbf{o}}_1^{T(k)}, \theta') \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} L(w_1^N, \hat{w}_1^{M(r)}) p(w_1^N | \hat{\mathbf{o}}_1^{T(r)}, \theta') \sum_{t=1}^T [\boldsymbol{\lambda}_{s_t}^\top \mathbf{f}(\hat{\mathbf{o}}_t) \\ & \quad + K(\boldsymbol{\lambda}_{s_t})] \end{aligned} \quad (\text{A.2.28})$$

$$\begin{aligned} & \frac{X_2(\theta, \theta')}{p(\hat{\mathbf{o}}_1^T | \theta')} \\ &= \prod_{k=1}^R \sum_{s_1^{T(k)} \in \mathcal{S}_k} p(s_1^{T(k)} | \hat{\mathbf{o}}_1^{T(k)}, \theta') \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} L(w_1^N, \hat{w}_1^{M(r)}) p(w_1^N | s_1^{T(r)}) \sum_{t=1}^T [\boldsymbol{\lambda}_{s_t}^\top \mathbf{f}(\hat{\mathbf{o}}_t) \\ & \quad + K(\boldsymbol{\lambda}_{s_t})] \end{aligned} \quad (\text{A.2.29})$$

The quantity $X_1(\theta, \theta')$ is now expressed as a sum over all states s . Note that Equation A.2.30 holds for any function $\mathbf{g}(\hat{\mathbf{o}}_t, s_t)$ of the t -th observation $\hat{\mathbf{o}}_t$ and t -th state s_t .

$$\begin{aligned}
& \prod_{k=1}^R \sum_{s_1^{T(k)} \in \mathcal{S}_k} p(s_1^{T(k)} | \hat{\mathbf{o}}_1^{T(k)}, \theta') \sum_{t=1}^T \mathbf{g}(\hat{\mathbf{o}}_t, s_t) \\
&= \sum_s \sum_{t=1}^T \prod_{k=1}^R \sum_{s_1^{T(k)} \in \mathcal{S}_k} p(s_1^{T(k)} | \hat{\mathbf{o}}_1^{T(k)}, \theta') 1_s(s_t) \mathbf{g}(\hat{\mathbf{o}}_t, s) \\
&= \sum_s \sum_{j=1}^R \sum_{t=1}^{T(j)} \prod_{k=1}^R \sum_{s_1^{T(k)} \in \mathcal{S}_k} p(s_1^{T(k)} | \hat{\mathbf{o}}_1^{T(k)}, \theta') 1_s(s_t(j)) \mathbf{g}(\hat{\mathbf{o}}_t(j), s) \quad (\text{A.2.30})
\end{aligned}$$

Here $s_t(j)$ is the t -th state in state sequence $s_1^{T(j)}$, $1_s(s_t)$ is the indicator function (equal to 1 if s_t is s and 0 otherwise), $\hat{\mathbf{o}}_t(j)$ is the t -th observation in sequence $\hat{\mathbf{o}}_1^{T(j)}$ and $T(j)$ is the length of state sequence $s_1^{T(j)}$. Note further that Equation A.2.31 holds.

$$\begin{aligned}
& \sum_{j=1}^R \sum_{t=1}^{T(j)} \prod_{k=1}^R \sum_{s_1^{T(k)} \in \mathcal{S}_k} p(s_1^{T(k)} | \hat{\mathbf{o}}_1^{T(k)}, \theta') 1_s(s_t(j)) \mathbf{g}(\hat{\mathbf{o}}_t(j), s) \\
&= \sum_{s_1^{T(1)} \in \mathcal{S}_1} p(s_1^{T(1)} | \hat{\mathbf{o}}_1^{T(1)}, \theta') \sum_{s_1^{T(2)} \in \mathcal{S}_2} p(s_1^{T(2)} | \hat{\mathbf{o}}_1^{T(2)}, \theta') \dots \sum_{s_1^{T(k)} \in \mathcal{S}_k} p(s_1^{T(k)} | \hat{\mathbf{o}}_1^{T(k)}, \theta') \dots \\
& \quad \dots \sum_{s_R \in \mathcal{S}_R} p(s_1^{T(R)} | \hat{\mathbf{o}}_1^{T(R)}, \theta') \sum_{j=1}^R \sum_{t=1}^{T(j)} 1_s(s_t(j)) \mathbf{g}(\hat{\mathbf{o}}_t(j), s) \\
&= \sum_{s_1^{T(1)} \in \mathcal{S}_1} p(s_1^{T(1)} | \hat{\mathbf{o}}_1^{T(1)}, \theta') \sum_{t=1}^{T(1)} 1_s(s_t(1)) \mathbf{g}(\hat{\mathbf{o}}_t(1), s) \\
& \quad + \sum_{s_1^{T(2)} \in \mathcal{S}_2} p(s_1^{T(2)} | \hat{\mathbf{o}}_1^{T(2)}, \theta') \sum_{t=1}^{T(2)} 1_s(s_t(2)) \mathbf{g}(\hat{\mathbf{o}}_t(2), s) + \dots \\
& \quad \dots + \sum_{s_1^{T(R)} \in \mathcal{S}_R} p(s_1^{T(R)} | \hat{\mathbf{o}}_1^{T(R)}, \theta') \sum_{t=1}^{T(R)} 1_s(s_t(R)) \mathbf{g}(\hat{\mathbf{o}}_t(R), s) \\
&= \sum_{j=1}^R \sum_{t=1}^{T(j)} \gamma_s(t, j) \mathbf{g}(\hat{\mathbf{o}}_t(j), s) \quad (\text{A.2.31})
\end{aligned}$$

Above $\gamma_s(t, j)$ is defined as the posterior probability, $p(s_t(j) = s | \hat{\mathbf{o}}_1^{T(j)}, \theta')$, that state s is the t -th state of state sequence $s_1^{T(j)}$.

Setting $\mathbf{g}(\hat{\mathbf{o}}_t(j), s_t) = \boldsymbol{\lambda}_{s_t}^\top \mathbf{f}(\hat{\mathbf{o}}_t) + K(\boldsymbol{\lambda}_{s_t})$ and using Equations A.2.30 and A.2.31, the quantity $X_1(\theta, \theta')$ is rephrased as shown by Equation A.2.32.

$$\begin{aligned}
& \frac{X_1(\theta, \theta')}{p(\hat{\mathbf{o}}_1^T | \theta')} \\
&= \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} L(w_1^N, \hat{w}_1^{M(r)}) p(w_1^N | \hat{\mathbf{o}}_1^{T(r)}, \theta') \prod_{k=1}^R \sum_{s_1^{T(k)} \in \mathcal{S}_k} p(s_1^{T(k)} | \hat{\mathbf{o}}_1^{T(k)}, \theta') \sum_{t=1}^T [\boldsymbol{\lambda}_{s_t}^\top \mathbf{f}(\hat{\mathbf{o}}_t) \\
& \hspace{15em} + K(\boldsymbol{\lambda}_{s_t})] \\
&= \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} L(w_1^N, \hat{w}_1^{M(r)}) p(w_1^N | \hat{\mathbf{o}}_1^{T(r)}, \theta') \sum_s \sum_{j=1}^R \sum_{t=1}^{T(j)} \gamma_s(t, j) [\boldsymbol{\lambda}_s^\top \mathbf{f}(\hat{\mathbf{o}}_t(j)) + K(\boldsymbol{\lambda}_s)]
\end{aligned} \tag{A.2.32}$$

Similarly, the quantity $X_2(\theta, \theta')$ (Equation A.2.29) is now expressed as a sum over all states. As beforehand, let $\mathbf{g}(\hat{\mathbf{o}}_t, s_t)$ be any function of $\hat{\mathbf{o}}_t$ and s_t . Note that Equation A.2.33 holds for any such function.

$$\begin{aligned}
& \prod_{k=1}^R \sum_{s_1^{T(k)} \in \mathcal{S}_k} p(s_1^{T(k)} | \hat{\mathbf{o}}_1^{T(k)}, \theta') p(w_1^N | s_1^{T(r)}) \sum_{t=1}^T \mathbf{g}(\hat{\mathbf{o}}_t, s_t) \\
&= \sum_s \sum_{t=1}^T \prod_{k=1}^R \sum_{s_1^{T(k)} \in \mathcal{S}_k} p(s_1^{T(k)} | \hat{\mathbf{o}}_1^{T(k)}, \theta') 1_s(s_t) p(w_1^N | s_1^{T(r)}) \mathbf{g}(\hat{\mathbf{o}}_t, s)
\end{aligned} \tag{A.2.33}$$

Expanding the right hand side of Equation A.2.33 gives Equation A.2.34.

$$\begin{aligned}
& \prod_{k=1}^R \sum_{s_1^{T(k)} \in \mathcal{S}_k} p(s_1^{T(k)} | \hat{\mathbf{o}}_1^{T(k)}, \theta') p(w_1^N | s_1^{T(r)}) \sum_{t=1}^T \mathbf{g}(\hat{\mathbf{o}}_t, s_t) \\
&= \sum_s \sum_{j=1}^R \sum_{t=1}^{T(j)} \prod_{k=1}^R \sum_{s_1^{T(k)} \in \mathcal{S}_k} p(s_1^{T(k)} | \hat{\mathbf{o}}_1^{T(k)}, \theta') 1_s(s_t(j)) p(w_1^N | s_1^{T(r)}) \mathbf{g}(\hat{\mathbf{o}}_t(j), s)
\end{aligned} \tag{A.2.34}$$

The inner terms of Equation A.2.34 are expanded in Equation A.2.35. The notation $\prod_{k \neq r}^R$

is shorthand for $\prod_{k=1, k \neq r}^R$, and the notation $\sum_{s_1^{T(k)}}$ is shorthand for $\sum_{s_1^{T(k)} \in \mathcal{S}_k}$.

$$\begin{aligned}
& \sum_{j=1}^R \sum_{t=1}^{T(j)} \prod_{k \neq r} \sum_{s_1^{T(k)}} p(s_1^{T(k)} | \hat{\mathbf{o}}_1^{T(k)}, \theta') \sum_{s_1^{T(r)}} p(w_1^N | s_1^{T(r)}) p(s_1^{T(r)} | \hat{\mathbf{o}}_1^{T(r)}, \theta') 1_s(s_t(j)) \mathbf{g}(\hat{\mathbf{o}}_t(j), s) \\
&= \prod_{k \neq r} \sum_{s_1^{T(k)}} p(s_1^{T(k)} | \hat{\mathbf{o}}_1^{T(k)}, \theta') \sum_{s_1^{T(r)}} p(s_1^{T(r)} | \hat{\mathbf{o}}_1^{T(r)}, \theta') p(w_1^N | s_1^{T(r)}) \sum_{j=1}^R \sum_{t=1}^{T(j)} 1_s(s_t(j)) \mathbf{g}(\hat{\mathbf{o}}_t(j), s) \\
&= \prod_{k \neq r} \sum_{s_1^{T(k)}} p(s_1^{T(k)} | \hat{\mathbf{o}}_1^{T(k)}, \theta') \sum_{s_1^{T(r)}} p(s_1^{T(r)} | \hat{\mathbf{o}}_1^{T(r)}, \theta') p(w_1^N | s_1^{T(r)}) \sum_{t=1}^{T(1)} 1_s(s_t(1)) \mathbf{g}(\hat{\mathbf{o}}_t(1), s) \\
&\quad + \prod_{k \neq r} \sum_{s_1^{T(k)}} p(s_1^{T(k)} | \hat{\mathbf{o}}_1^{T(k)}, \theta') \sum_{s_1^{T(r)}} p(s_1^{T(r)} | \hat{\mathbf{o}}_1^{T(r)}, \theta') p(w_1^N | s_1^{T(r)}) \sum_{t=1}^{T(2)} 1_s(s_t(2)) \mathbf{g}(\hat{\mathbf{o}}_t(2), s) \\
&\quad + \dots \\
&\quad + \prod_{k \neq r} \sum_{s_1^{T(k)}} p(s_1^{T(k)} | \hat{\mathbf{o}}_1^{T(k)}, \theta') \sum_{s_1^{T(r)}} p(s_1^{T(r)} | \hat{\mathbf{o}}_1^{T(r)}, \theta') p(w_1^N | s_1^{T(r)}) \sum_{t=1}^{T(r)} 1_s(s_t(r)) \mathbf{g}(\hat{\mathbf{o}}_t(r), s) \\
&\quad + \dots \\
&\quad + \prod_{k \neq r} \sum_{s_1^{T(k)}} p(s_1^{T(k)} | \hat{\mathbf{o}}_1^{T(k)}, \theta') \sum_{s_1^{T(r)}} p(s_1^{T(r)} | \hat{\mathbf{o}}_1^{T(r)}, \theta') p(w_1^N | s_1^{T(r)}) \sum_{t=1}^{T(R)} 1_s(s_t(R)) \mathbf{g}(\hat{\mathbf{o}}_t(R), s)
\end{aligned} \tag{A.2.35}$$

Equation A.2.35 simplifies to Equation A.2.36.

$$\begin{aligned}
& \sum_{j=1}^R \sum_{t=1}^{T(j)} \prod_{k \neq r} \sum_{s_1^{T(k)}} p(s_1^{T(k)} | \hat{\mathbf{o}}_1^{T(k)}, \theta') \sum_{s_1^{T(r)}} p(w_1^N | s_1^{T(r)}) p(s_1^{T(r)} | \hat{\mathbf{o}}_1^{T(r)}, \theta') 1_s(s_t(j)) \mathbf{g}(\hat{\mathbf{o}}_t(j), s) \\
&= \sum_{s_1^{T(1)}} p(s_1^{T(1)} | \hat{\mathbf{o}}_1^{T(1)}, \theta') \sum_{s_1^{T(r)}} p(s_1^{T(r)} | \hat{\mathbf{o}}_1^{T(r)}, \theta') p(w_1^N | s_1^{T(r)}) \sum_{t=1}^{T(1)} 1_s(s_t(1)) \mathbf{g}(\hat{\mathbf{o}}_t(1), s) \\
&\quad + \dots \\
&\quad + \sum_{s_1^{T(r)}} p(s_1^{T(r)} | \hat{\mathbf{o}}_1^{T(r)}, \theta') p(w_1^N | s_1^{T(r)}) \sum_{t=1}^{T(r)} 1_s(s_t(r)) \mathbf{g}(\hat{\mathbf{o}}_t(r), s) \\
&\quad + \dots \\
&\quad + \sum_{s_1^{T(R)}} p(s_1^{T(R)} | \hat{\mathbf{o}}_1^{T(R)}, \theta') \sum_{s_1^{T(r)}} p(s_1^{T(r)} | \hat{\mathbf{o}}_1^{T(r)}, \theta') p(w_1^N | s_1^{T(r)}) \sum_{t=1}^{T(R)} 1_s(s_t(R)) \mathbf{g}(\hat{\mathbf{o}}_t(R), s)
\end{aligned} \tag{A.2.36}$$

Since $p(w_1^N | s_1^{T(r)})$ is equal to $p(w_1^N | s_1^{T(r)}, \hat{\mathbf{o}}_1^{T(r)}, \theta')$ (the word sequence w_1^N is conditionally independent of $\hat{\mathbf{o}}_1^{T(r)}$ and θ' , given the state sequence $s_1^{T(r)}$), Equation A.2.34 simplifies further by using Equation A.2.36 to yield Equation A.2.37.

$$\begin{aligned}
& \prod_{k=1}^R \sum_{s_1^{T(k)} \in \mathcal{S}_k} p(s_1^{T(k)} | \hat{\mathbf{o}}_1^{T(k)}, \theta') p(w_1^N | s_1^{T(r)}) \sum_{t=1}^T \mathbf{g}(\hat{\mathbf{o}}_t, s_t) \\
= & p(w_1^N | \hat{\mathbf{o}}_1^{T(r)}, \theta') \sum_s \sum_{s_1^{T(1)} \in \mathcal{S}_1} p(s_1^{T(1)} | \hat{\mathbf{o}}_1^{T(1)}, \theta') \sum_{t=1}^{T(1)} 1_s(s_t(1)) \mathbf{g}(\hat{\mathbf{o}}_t(1), s) \\
& + p(w_1^N | \hat{\mathbf{o}}_1^{T(r)}, \theta') \sum_s \sum_{s_1^{T(2)} \in \mathcal{S}_1} p(s_1^{T(2)} | \hat{\mathbf{o}}_1^{T(2)}, \theta') \sum_{t=1}^{T(2)} 1_s(s_t(2)) \mathbf{g}(\hat{\mathbf{o}}_t(2), s) + \dots \\
& + p(w_1^N | \hat{\mathbf{o}}_1^{T(r)}, \theta') \sum_s \sum_{s_1^{T(r)} \in \mathcal{S}_r} p(s_1^{T(r)} | w_1^N, \hat{\mathbf{o}}_1^{T(r)}, \theta') \sum_{t=1}^{T(r)} 1_s(s_t(r)) \mathbf{g}(\hat{\mathbf{o}}_t(r), s) + \dots \\
& + p(w_1^N | \hat{\mathbf{o}}_1^{T(r)}, \theta') \sum_s \sum_{s_1^{T(R)} \in \mathcal{S}_R} p(s_1^{T(R)} | \hat{\mathbf{o}}_1^{T(R)}, \theta') \sum_{t=1}^{T(R)} 1_s(s_t(R)) \mathbf{g}(\hat{\mathbf{o}}_t(R), s) \\
= & p(w_1^N | \hat{\mathbf{o}}_1^{T(r)}, \theta') \sum_s \left[\sum_{j=1, j \neq r}^R \sum_{t=1}^{T(j)} \gamma_s(t, j) \mathbf{g}(\hat{\mathbf{o}}_t(j), s) + \sum_{t=1}^{T(r)} \gamma_s(t, r, w_1^N) \mathbf{g}(\hat{\mathbf{o}}_t(r), s) \right]
\end{aligned} \tag{A.2.37}$$

The quantity $\gamma_s(t, r, w_1^N)$ is the posterior probability, $p(s_t(r) = s | \hat{\mathbf{o}}_1^{T(r)}, w_1^N, \theta')$, that state s is the t -th state of state sequence $s_1^{T(r)}$, given word sequence w_1^N .

Using Equation A.2.37 with $\mathbf{g}(\hat{\mathbf{o}}_t, s_t) = \boldsymbol{\lambda}_{s_t}^T \mathbf{f}(\hat{\mathbf{o}}_t) + K(\boldsymbol{\lambda}_{s_t})$, the quantity $X_2(\theta, \theta')$ (see Equation A.2.29) is expressed by Equation A.2.38.

$$\begin{aligned}
& \frac{X_2(\theta, \theta')}{p(\hat{\mathbf{o}}_1^T | \theta')} \\
= & \sum_s \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} L(w_1^N, \hat{w}_1^{M(r)}) p(w_1^N | \hat{\mathbf{o}}_1^{T(r)}, \theta') \left[\sum_{j=1, j \neq r}^R \sum_{t=1}^{T(j)} \gamma_s(t, j) \left[\boldsymbol{\lambda}_s^T \mathbf{f}(\hat{\mathbf{o}}_t(j)) \right. \right. \\
& \left. \left. + K(\boldsymbol{\lambda}_s) \right] + \sum_{t=1}^{T(r)} \gamma_s(t, r, w_1^N) \left[\boldsymbol{\lambda}_s^T \mathbf{f}(\hat{\mathbf{o}}_t(j)) + K(\boldsymbol{\lambda}_s) \right] \right]
\end{aligned} \tag{A.2.38}$$

Using Equations A.2.39 and A.2.40 yields Equation A.2.41, where Equations A.2.14 and A.2.15 are also used.

$$\begin{aligned}
& H(\theta, \theta') \\
&= X_1(\theta, \theta') - X_2(\theta, \theta') + Y(\theta, \theta') \\
&= p(\hat{\mathbf{o}}_1^T | \theta') \sum_s \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} L(w_1^N, \hat{w}_1^{M(r)}) p(w_1^N | \hat{\mathbf{o}}_1^{T(r)}, \theta') \left[\sum_{t=1}^{T(r)} \left[\gamma_s(t, r) \right. \right. \\
&\qquad \qquad \qquad \qquad \qquad \qquad \qquad \qquad \qquad \qquad \qquad \qquad \qquad \qquad \qquad \qquad \qquad \left. \left. - \gamma_s(t, r, w_1^N) \right] \left[\boldsymbol{\lambda}_s^T \mathbf{f}(\hat{\mathbf{o}}_t(j)) + K(\boldsymbol{\lambda}_s) \right] \right] \\
&\quad + \sum_s D_s \boldsymbol{\lambda}_s^T E_{\theta'_s} \{ \mathbf{f}(\mathbf{o}_1^T) \} + \sum_s D_s K(\boldsymbol{\lambda}_s) \\
&= \sum_s \left[\gamma_s K(\boldsymbol{\lambda}_s) + \boldsymbol{\lambda}_s^T \boldsymbol{\Gamma}_s \right] \tag{A.2.41}
\end{aligned}$$

It has now been shown that Equation A.2.13 holds. The next section demonstrates that $F_{\text{MBR}}(\theta, \theta', D)$ is an auxiliary function for the MBR criterion $R_{\text{MBR}}(\theta)$.

A.3 Proof of validity of auxiliary function

Similar arguments which prove that the auxiliary function for the conditional ML criterion is valid (Axelrod et al. (2007)) hold in the case of the defined auxiliary function for the MBR criterion. This proof is given below, in the form of three theorems. For the purposes of all three theorems, let θ represent the acoustic model parameters of the means and covariances of each mixture component in a continuous density HMM system. Further, let Θ represent the set of all such parameters and let θ' represent the initial parameters.

Theorem 3. Fix positive constants $d(s_1^T)$ in Equation A.2.2, and then define the following for some positive real number ϵ .

$$d'(s_1^T) = \frac{d(s_1^T)}{\epsilon} \tag{A.3.1}$$

$$q_0(s_1^T) = d(s_1^T) \tag{A.3.2}$$

$$q'(\hat{\mathbf{o}}_1^T, s_1^T) = \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} 1_{\hat{\mathbf{o}}_1^T}(\mathbf{o}_1^T) L(w_1^N, \hat{w}_1^{M(r)}) \left[p(w_1^N | \hat{\mathbf{o}}_1^{T(r)}, \theta') - p(w_1^N | s_1^{T(r)}) \right] \tag{A.3.3}$$

$$q_\epsilon(\hat{\mathbf{o}}_1^T, s_1^T) = q_0(s_1^T) + \epsilon q'(\hat{\mathbf{o}}_1^T, s_1^T) \tag{A.3.4}$$

$$Q_\epsilon(\theta) = \int_{\mathbf{o}_1^T} \sum_{s_1^T \in \mathcal{S}} p(\mathbf{o}_1^T, s_1^T | \theta) q_\epsilon(\hat{\mathbf{o}}_1^T, s_1^T) \log \frac{p(\mathbf{o}_1^T, s_1^T | \theta)}{p(\mathbf{o}_1^T, s_1^T | \theta')} d\mathbf{o}_1^T \tag{A.3.5}$$

Then it can be shown that Equation A.3.6 holds, where D' is defined by Equation A.3.7 and $F_{\text{MBR}}(\theta, \theta', D')$ is defined by Equation A.2.1.

$$\frac{Q_\epsilon(\theta)}{\epsilon} = F_{\text{MBR}}(\theta, \theta', D') - F_{\text{MBR}}(\theta', \theta', D') \quad (\text{A.3.6})$$

$$D' = \sum_{s_1^T \in \mathcal{S}} p(s_1^T) d'(s_1^T) \quad (\text{A.3.7})$$

Suppose the hypothesis space \mathcal{W} of Equation A.0.1 is finite. Then, given a sufficiently small value of ϵ , the subset of Θ defined by $\{\theta \in \Theta : Q_\epsilon(\theta) > 0\}$ is bounded.

Theorem 4. Define the function $x(\mathbf{o}_1^T, s_1^T, \theta', \theta)$ as in Equation A.3.8, where θ and θ' are members of a compact set κ , $y(\mathbf{o}_1^T, s_1^T, \theta', \theta)$ is defined by Equation A.3.9 and $z(a)$ is defined by Equation A.3.10.

$$x(\mathbf{o}_1^T, s_1^T, \theta', \theta) = p(\mathbf{o}_1^T, s_1^T | \theta') y(\mathbf{o}_1^T, s_1^T, \theta', \theta) \quad (\text{A.3.8})$$

$$y(\mathbf{o}_1^T, s_1^T, \theta', \theta) = z\left(\frac{p(\mathbf{o}_1^T, s_1^T | \theta)}{p(\mathbf{o}_1^T, s_1^T | \theta')} - 1\right) \quad (\text{A.3.9})$$

$$z(a) = a - \log(1 + a) \quad (\text{A.3.10})$$

Then there is a constant $C > 0$ which is independent of θ , such that Equation A.3.11 holds.

$$\int_{\mathbf{o}_1^T} x(\mathbf{o}_1^T, s_1^T, \theta', \theta) d\mathbf{o}_1^T \geq C x(\hat{\mathbf{o}}_1^T, s_1^T, \theta', \theta) \quad (\text{A.3.11})$$

Theorem 5. Suppose the hypothesis space \mathcal{W} of Equation A.0.1 is finite. Let κ_ϵ be the subset of Θ for which $Q_\epsilon(\theta) > 0$. Then there exists a positive ϵ_1 such that for any positive ϵ less than ϵ_1 , and $\theta \in \kappa_\epsilon$, the implication of Equation A.3.12 holds.

$$Q_\epsilon(\theta) > 0 \Rightarrow R_{\text{MBR}}(\theta') - R_{\text{MBR}}(\theta) \geq 0 \quad (\text{A.3.12})$$

This theorem, in combination with Equation A.3.6, shows that $F_{\text{MBR}}(\theta, \theta', D')$ is an auxiliary function for $R_{\text{MBR}}(\theta)$ provided ϵ is sufficiently small (or equivalently, each $d'(s_1^T)$ is sufficiently large).

A.3.1 Preliminary results

Before embarking upon the proofs of the above theorems, some preliminary lemmas are stated and proved where necessary. These results are used later in the proof of Theorem 3.

Lemma 6. Suppose \mathbf{A} is a positive definite symmetric real $d \times d$ matrix. Then $\mathbf{A} = \mathbf{P}^{-1} \mathbf{D} \mathbf{P}$ where \mathbf{D} is a diagonal $d \times d$ matrix with diagonal entries which are all positive and \mathbf{P} is a unitary $d \times d$ matrix. The diagonal entries of \mathbf{D} are the eigenvalues of \mathbf{A} .

Lemma 7. Let \mathbf{A} represent a positive definite symmetric real $d \times d$ matrix and let \mathbf{x} and \mathbf{y} represent $d \times 1$ vectors. Then $\mathbf{x}^\top \mathbf{A} \mathbf{y} \leq |\mathbf{x}| |\mathbf{y}| \text{tr}(\mathbf{A})$ with equality if $\mathbf{x} = \mathbf{y}$. Here $\text{tr}(\mathbf{A})$ is the trace of \mathbf{A} .

Lemma 8. Let \mathbf{A} represent a positive definite symmetric real $d \times d$ matrix. Then $\log \det \mathbf{A} \leq \text{tr}(\mathbf{A}) - d$.

Lemma 9. Let \mathbf{A} , \mathbf{B} and \mathbf{C} represent $d \times d$ matrices. Then $\text{tr}(\mathbf{ABC}) = \text{tr}(\mathbf{CAB}) = \text{tr}(\mathbf{BCA})$.

Lemma 10. Let \mathbf{A} represent a $d \times d$ matrix and let \mathbf{B} represent an invertible $d \times d$ matrix. Then $\text{tr}(\mathbf{B}^{-1}\mathbf{AB}) = \text{tr}(\mathbf{A})$.

Lemma 11. Let \mathbf{A} represent a positive definite symmetric real $d \times d$ matrix. Then $\text{tr}(\mathbf{A}) = \text{tr}(\mathbf{P}^{-1}\mathbf{DP}) = \text{tr}(\mathbf{D})$, where \mathbf{D} is a diagonal $d \times d$ matrix with diagonal entries which are all positive and \mathbf{P} is a unitary $d \times d$ matrix.

Lemma 12. Let \mathbf{A} represent a positive definite symmetric real $d \times d$ matrix and let \mathbf{B} represent a symmetric $d \times d$ matrix matrix. Then $\text{tr}(\mathbf{AB}) \leq \text{tr}(\mathbf{A})\text{tr}(\mathbf{B})$.

Proof of Lemma 12

To prove Lemma 12, firstly note that, by Lemma 6, Equation A.3.13 holds for a unitary matrix \mathbf{P} and a diagonal matrix \mathbf{D} whose diagonal entries are all positive.

$$\text{tr}(\mathbf{AB}) = \text{tr}(\mathbf{P}^{-1}\mathbf{DPB}) \quad (\text{A.3.13})$$

By the cyclic property of the trace (Lemma 9), the inequality of Equation A.3.14 is derived from Equation A.3.13, where $\text{diag}(\mathbf{X})$ represents the diagonal vector of matrix \mathbf{X} .

$$\begin{aligned} \text{tr}(\mathbf{AB}) &= \text{tr}(\mathbf{PBP}^{-1}\mathbf{D}) \\ &= (\text{diag}(\mathbf{PBP}^{-1}))^T \text{diag}(\mathbf{D}) \\ &\leq |\text{diag}(\mathbf{PBP}^{-1})| |\text{diag}(\mathbf{D})| \end{aligned} \quad (\text{A.3.14})$$

Since $|\text{diag}(\mathbf{D})| \leq \text{tr}(\mathbf{D})$ and by Lemma 11, $\text{tr}(\mathbf{D}) = \text{tr}(\mathbf{A})$, the inequality of Equation A.3.15 holds true.

$$\text{tr}(\mathbf{AB}) \leq |\text{diag}(\mathbf{PBP}^{-1})| \text{tr}(\mathbf{A}) \quad (\text{A.3.15})$$

Note that $\mathbf{PBP}^{-1} = \mathbf{PLL}^*\mathbf{P}^{-1} = \mathbf{PL}(\mathbf{PL})^*$, where \mathbf{LL}^* is the Choleski decomposition of symmetric matrix \mathbf{B} and \mathbf{X}^* represents the conjugate transpose of \mathbf{X} . This decomposition shows that the diagonal entries of \mathbf{PBP}^{-1} are all non-negative, so the inequality of Equation A.3.16 follows.

$$|\text{diag}(\mathbf{PBP}^{-1})| \leq \text{tr}(\mathbf{PBP}^{-1}) \quad (\text{A.3.16})$$

By the similarity property of the trace operator (Lemma 10), $\text{tr}(\mathbf{PBP}^{-1}) = \text{tr}(\mathbf{B})$ and Lemma 12 follows.

Lemma 13. Define the spectral norm $\|\mathbf{A}\|_{sp}$ of \mathbf{A} as the square root of the largest eigenvalue of the positive semi-definite matrix $\mathbf{A}^*\mathbf{A}$. Then the spectral norm of a positive definite matrix is the largest eigenvalue of the matrix.

Lemma 14. Define the max norm $\|\mathbf{A}\|_{max}$ of an $m \times n$ matrix \mathbf{A} as $\max_{i \leq m, j \leq n} |a_{ij}|$, where a_{ij} is the element in the i -th row and the j -th column. Then the max norm and the spectral norm are equivalent, i.e. there exist $r > 0$ and $s > 0$ such that $r\|\mathbf{A}\|_{max} \leq \|\mathbf{A}\|_{sp} \leq s\|\mathbf{A}\|_{max}$.

A.3.2 Proof of Theorem 3

Equations A.3.6 and A.2.10 give Equation A.3.17.

$$\begin{aligned} \frac{Q_\epsilon(\theta)}{\epsilon} &= F_{\text{MBR}}(\theta, \theta', D') - F_{\text{MBR}}(\theta', \theta', D') \\ &= H(\theta, \theta') - H(\theta', \theta') \end{aligned} \quad (\text{A.3.17})$$

Using Equation A.2.13 gives Equation A.3.18. The symbol $\boldsymbol{\lambda}_s$ represents the vector of Equation A.2.7 given the parameter set θ . The symbol $\boldsymbol{\lambda}'_s$ represents the vector of Equation A.2.7 given the parameter set θ' .

$$\begin{aligned} \frac{Q_\epsilon(\theta)}{\epsilon} &= H(\theta, \theta') - H(\theta', \theta') \\ &= \sum_s \left[\gamma_s K(\boldsymbol{\lambda}_s) + \boldsymbol{\lambda}_s^\top \boldsymbol{\Gamma}_s \right] - \sum_s \left[\gamma_s K(\boldsymbol{\lambda}'_s) + (\boldsymbol{\lambda}'_s)^\top \boldsymbol{\Gamma}_s \right] \end{aligned} \quad (\text{A.3.18})$$

Given Equation A.3.18, $Q_\epsilon(\theta)$ is expressed by Equation A.3.19 where $h_s(\theta) - h_s(\theta')$ is expressed by Equation A.3.20.

$$Q_\epsilon(\theta) = \sum_s [h_s(\theta) - h_s(\theta')] \quad (\text{A.3.19})$$

$$\begin{aligned} h_s(\theta) - h_s(\theta') &= \epsilon \left[\gamma_s K(\boldsymbol{\lambda}_s) + \boldsymbol{\lambda}_s^\top \boldsymbol{\Gamma}_s \right] - \left[\gamma_s K(\boldsymbol{\lambda}'_s) + (\boldsymbol{\lambda}'_s)^\top \boldsymbol{\Gamma}_s \right] \\ &= \epsilon \left[\gamma_s (K(\boldsymbol{\lambda}_s) - K(\boldsymbol{\lambda}'_s)) + (\boldsymbol{\lambda}_s^\top - (\boldsymbol{\lambda}'_s)^\top) \boldsymbol{\Gamma}_s \right] \end{aligned} \quad (\text{A.3.20})$$

Let $\boldsymbol{\Gamma}_s = \begin{bmatrix} \text{vec}(\mathbf{A}_s) \\ \mathbf{b}_s \end{bmatrix}$, where \mathbf{A}_s is symmetric, of dimension $d \times d$ and \mathbf{b}_s is a column vector of dimension d , where d is the dimensionality of the mean vector $\boldsymbol{\mu}_s$. Equation A.3.20 may be rewritten as Equation A.3.21. The symbols $\boldsymbol{\mu}_s$ and \mathbf{C}_s represent the mean and covariance of state s given parameter set θ , while $\boldsymbol{\mu}'_s$ and \mathbf{C}'_s represent the mean and covariance of state s given parameter set θ' .

$$\begin{aligned} h_s(\theta) - h_s(\theta') &= \epsilon \left[\gamma_s (K(\boldsymbol{\lambda}_s) - K(\boldsymbol{\lambda}'_s)) + (\boldsymbol{\lambda}_s^\top - (\boldsymbol{\lambda}'_s)^\top) \boldsymbol{\Gamma}_s \right] \\ &= \epsilon \left[\gamma_s (K(\boldsymbol{\lambda}_s) - K(\boldsymbol{\lambda}'_s)) + [\text{vec}(\mathbf{C}_s^{-1}) - \text{vec}((\mathbf{C}'_s)^{-1})]^\top \text{vec}(\mathbf{A}_s) \right. \\ &\quad \left. + [\mathbf{C}_s^{-1} \boldsymbol{\mu}_s - (\mathbf{C}'_s)^{-1} \boldsymbol{\mu}'_s]^\top \mathbf{b}_s \right] \\ &= \epsilon \left[\gamma_s (K(\boldsymbol{\lambda}_s) - K(\boldsymbol{\lambda}'_s)) + \text{tr}(\mathbf{C}_s^{-1} \mathbf{A}_s) - \text{tr}((\mathbf{C}'_s)^{-1} \mathbf{A}_s) \right. \\ &\quad \left. + \boldsymbol{\mu}_s^\top \mathbf{C}_s^{-1} \mathbf{b}_s - (\boldsymbol{\mu}'_s)^\top (\mathbf{C}'_s)^{-1} \mathbf{b}_s \right] \end{aligned} \quad (\text{A.3.21})$$

Using Equation A.2.9 and Lemmas 7 and 8, $K(\boldsymbol{\lambda}_s) - K(\boldsymbol{\lambda}'_s)$ is bounded above as shown by Equation A.3.22. The fact that covariance matrices are positive definite and symmetric has been used.

$$\begin{aligned} K(\boldsymbol{\lambda}_s) - K(\boldsymbol{\lambda}'_s) &= \frac{1}{2} \left[\log \det(\mathbf{C}_s^{-1}) - \log \det((\mathbf{C}'_s)^{-1}) - [(\boldsymbol{\mu}_s)^\top \mathbf{C}_s^{-1} \boldsymbol{\mu}_s - (\boldsymbol{\mu}'_s)^\top (\mathbf{C}'_s)^{-1} \boldsymbol{\mu}'_s] \right] \\ &\leq \frac{1}{2} \left[\text{tr}(\mathbf{C}_s^{-1}) - d - \log \det((\mathbf{C}'_s)^{-1}) + \text{tr}((\mathbf{C}'_s)^{-1}) |\boldsymbol{\mu}'_s|^2 - \text{tr}(\mathbf{C}_s^{-1}) |\boldsymbol{\mu}_s|^2 \right] \end{aligned} \quad (\text{A.3.22})$$

Using the inequality of Equation A.3.22 in Equation A.3.21 gives the inequality of Equation A.3.23, where Lemmas 7 and 12 have also been applied, using the fact that covariance matrices are positive definite and symmetric.

$$\begin{aligned} &h_s(\theta) - h_s(\theta') \\ &\leq \epsilon \left[\frac{\gamma_s}{2} \left[\text{tr}(\mathbf{C}_s^{-1}) - d - \log \det((\mathbf{C}'_s)^{-1}) + \text{tr}((\mathbf{C}'_s)^{-1}) |\boldsymbol{\mu}'_s|^2 - \text{tr}(\mathbf{C}_s^{-1}) |\boldsymbol{\mu}_s|^2 \right] \right. \\ &\quad \left. - \text{tr}((\mathbf{C}'_s)^{-1} \mathbf{A}_s) + \text{tr}(\mathbf{C}_s^{-1} \mathbf{A}_s) \right. \\ &\quad \left. + \text{tr}(\mathbf{C}_s^{-1}) |\boldsymbol{\mu}_s| |\mathbf{b}_s| - (\boldsymbol{\mu}'_s)^\top (\mathbf{C}'_s)^{-1} \mathbf{b}_s \right] \\ &\leq \epsilon \left[\frac{\gamma_s}{2} \left[\text{tr}(\mathbf{C}_s^{-1}) - d - \log \det((\mathbf{C}'_s)^{-1}) + \text{tr}((\mathbf{C}'_s)^{-1}) |\boldsymbol{\mu}'_s|^2 - \text{tr}(\mathbf{C}_s^{-1}) |\boldsymbol{\mu}_s|^2 \right] \right. \\ &\quad \left. - \text{tr}((\mathbf{C}'_s)^{-1} \mathbf{A}_s) + \text{tr}(\mathbf{C}_s^{-1}) \text{tr}(\mathbf{A}_s) \right. \\ &\quad \left. + \text{tr}(\mathbf{C}_s^{-1}) |\boldsymbol{\mu}_s| |\mathbf{b}_s| - (\boldsymbol{\mu}'_s)^\top (\mathbf{C}'_s)^{-1} \mathbf{b}_s \right] \end{aligned} \quad (\text{A.3.23})$$

Let a_s be defined by Equation A.3.24.

$$\begin{aligned} a_s &= \frac{\gamma_s}{2} \left[-d - \log \det((\mathbf{C}'_s)^{-1}) + \text{tr}((\mathbf{C}'_s)^{-1}) |\boldsymbol{\mu}'_s|^2 \right] \\ &\quad - \text{tr}((\mathbf{C}'_s)^{-1} \mathbf{A}_s) - (\boldsymbol{\mu}'_s)^\top (\mathbf{C}'_s)^{-1} \mathbf{b}_s \end{aligned} \quad (\text{A.3.24})$$

The inequality of Equation A.3.23 may be rewritten as Equation A.3.25.

$$\begin{aligned} &h_s(\theta) - h_s(\theta') \\ &\leq \epsilon \left[\frac{\gamma_s}{2} \left[\text{tr}(\mathbf{C}_s^{-1}) - \text{tr}(\mathbf{C}_s^{-1}) |\boldsymbol{\mu}_s|^2 \right] + \text{tr}(\mathbf{C}_s^{-1}) \text{tr}(\mathbf{A}_s) + \text{tr}(\mathbf{C}_s^{-1}) |\boldsymbol{\mu}_s| |\mathbf{b}_s| + a_s \right] \\ &= \epsilon \left[\text{tr}(\mathbf{C}_s^{-1}) \left[\frac{\gamma_s}{2} (1 - |\boldsymbol{\mu}_s|^2) + \text{tr}(\mathbf{A}_s) + |\boldsymbol{\mu}_s| |\mathbf{b}_s| \right] + a_s \right] \end{aligned} \quad (\text{A.3.25})$$

The inequality of Equation A.3.25 may be rephrased as Equation A.3.26.

$$\begin{aligned} h_s(\theta) - h_s(\theta') &\leq \epsilon \left[\text{tr}(\mathbf{C}_s^{-1}) \left[\left(\frac{\gamma_s}{2} + |\boldsymbol{\mu}_s| |\mathbf{b}_s| \right) - \left(\frac{\gamma_s}{2} |\boldsymbol{\mu}_s|^2 - \text{tr}(\mathbf{A}_s) \right) \right] + a_s \right] \end{aligned} \quad (\text{A.3.26})$$

Suppose that $\sum_s [h_s(\theta) - h_s(\theta')] > 0$. Then the inequality of Equation A.3.27 holds.

$$\sum_s \text{tr}(\mathbf{C}_s^{-1}) \left[\left(\frac{\gamma_s}{2} |\boldsymbol{\mu}_s|^2 - \text{tr}(\mathbf{A}_s) \right) - \left(\frac{\gamma_s}{2} + |\boldsymbol{\mu}_s| |\mathbf{b}_s| \right) \right] \leq \sum_s a_s \quad (\text{A.3.27})$$

Since \mathcal{W} is finite, a sufficiently small ϵ may be chosen to ensure that $\gamma_s > 0$ and $\text{tr}(\mathbf{A}_s) < 0$ for all s . The inequality of Equation A.3.28 is then derived from Equation A.3.27.

$$\sum_s \text{tr}(\mathbf{C}_s^{-1}) \left[2\sqrt{\frac{\gamma_s}{2} |\boldsymbol{\mu}_s|^2 - \text{tr}(\mathbf{A}_s)} \sqrt{\frac{\gamma_s}{2} + |\boldsymbol{\mu}_s| |\mathbf{b}_s|} \right] \leq \sum_s a_s \quad (\text{A.3.28})$$

The inequality of Equation A.3.28 shows that both $\text{tr}(\mathbf{C}_s^{-1})$ and $|\boldsymbol{\mu}_s|$ are bounded above for all s . Since $\text{tr}(\mathbf{C}_s^{-1})$ is bounded, and is equal to the sum of the eigenvalues of the positive definite matrix \mathbf{C}_s^{-1} , the spectral norm of \mathbf{C}_s^{-1} is bounded (Lemma 13). Since the spectral norm and the max norm of a matrix are equivalent (Lemma 14), each element of the set of $n(n+1)/2$ parameters governing the inverse covariance matrix \mathbf{C}_s^{-1} is bounded above for all s . Hence $\{\theta \in \Theta : \frac{Q_\epsilon(\theta)}{\epsilon} > 0\}$ is bounded and Theorem 3 is proved.

A.3.3 Proof of Theorem 4

The proof of Theorem 4 is provided in Axelrod et al. (2007), and uses the facts that the function $x(\mathbf{o}_1^T, s_1^T, \theta', \theta)$ is a non-negative analytic function and that κ is compact.

A.3.4 Proof of Theorem 5

Firstly define

$$F_\epsilon(\theta) = \int_{\mathbf{o}_1^T} \sum_{s_1^T \in \mathcal{S}} q_\epsilon(\mathbf{o}_1^T, s_1^T) [p(\mathbf{o}_1^T, s_1^T | \theta') - p(\mathbf{o}_1^T, s_1^T | \theta)] d\mathbf{o}_1^T \quad (\text{A.3.29})$$

where $q_\epsilon(\hat{\mathbf{o}}_1^T, s_1^T)$ is defined in Equation A.3.4. Then Equation A.3.30 holds, where $F(\theta|\theta')$ is defined as $-G(\theta|\theta')$, and $G(\theta|\theta')$ is defined in Lemma 2.

$$\frac{F_\epsilon(\theta)}{\epsilon} = F(\theta'|\theta') - F(\theta|\theta') \quad (\text{A.3.30})$$

It has been shown in Lemma 2 that $F(\theta|\theta')$ is an auxiliary function for the MBR criterion $R_{\text{MBR}}(\theta)$, in the sense that, if $F(\theta|\theta')$ decreases, the MBR criterion does not increase. It therefore suffices to show that $Q_\epsilon(\theta) \leq F_\epsilon(\theta)$ for $\theta \in \kappa$. This is because the implication of Equation A.3.31 holds.

$$\begin{aligned} 0 < Q_\epsilon(\theta) \leq F_\epsilon(\theta) &\Rightarrow 0 < \epsilon(F(\theta'|\theta') - F(\theta|\theta')) \\ &\Rightarrow 0 < F(\theta'|\theta') - F(\theta|\theta') \\ &\Rightarrow 0 \leq R_{\text{MBR}}(\theta') - R_{\text{MBR}}(\theta) \end{aligned} \quad (\text{A.3.31})$$

Define $\Delta_\epsilon(\theta)$ as in Equation A.3.32.

$$\Delta_\epsilon(\theta) = F_\epsilon(\theta) - Q_\epsilon(\theta) \quad (\text{A.3.32})$$

Then it suffices to show that $\Delta_\epsilon(\theta)$ is non-negative for $\theta \in \kappa_\epsilon$. Begin by rewriting Equation A.3.32 as Equation A.3.33, where $q_\epsilon(\hat{\mathbf{o}}_1^T, s_1^T)$ is defined in Equation A.3.4 and $y(\mathbf{o}_1^T, s_1^T, \theta', \theta)$ is defined in Equation A.3.9.

$$\begin{aligned} \Delta_\epsilon(\theta) &= \sum_{s_1^T \in \mathcal{S}} \int_{\mathbf{o}_1^T} q_\epsilon(\hat{\mathbf{o}}_1^T, s_1^T) p(\mathbf{o}_1^T, s_1^T | \theta') y(\mathbf{o}_1^T, s_1^T, \theta', \theta) d\mathbf{o}_1^T \\ &= \Delta_0(\theta) + \epsilon \Delta'(\theta) \end{aligned} \quad (\text{A.3.33})$$

The term $\Delta'(\theta)$ is defined in Equation A.3.34 (where $q'(\hat{\mathbf{o}}_1^T, s_1^T)$ is defined in Equation A.3.3) and $\Delta_0(\theta)$ is defined in Equation A.3.35 (where $q_0(s_1^T)$ is defined in Equation A.3.2).

$$\begin{aligned} \Delta'(\theta) &= \sum_{s_1^T \in \mathcal{S}} \int_{\mathbf{o}_1^T} q'(\hat{\mathbf{o}}_1^T, s_1^T) p(\mathbf{o}_1^T, s_1^T | \theta') y(\mathbf{o}_1^T, s_1^T, \theta', \theta) d\mathbf{o}_1^T \\ &= \sum_{s_1^T \in \mathcal{S}} \int_{\mathbf{o}_1^T} \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} \mathbf{1}_{\hat{\mathbf{o}}_1^T} L(w_1^N, \hat{w}_1^{M(r)}) [p(w_1^N | \hat{\mathbf{o}}_1^{T(r)}, \theta') \\ &\quad - p(w_1^N | s_1^{T(r)})] p(\mathbf{o}_1^T, s_1^T | \theta') y(\mathbf{o}_1^T, s_1^T, \theta', \theta) d\mathbf{o}_1^T \\ &= \sum_{s_1^T \in \mathcal{S}} \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} L(w_1^N, \hat{w}_1^{M(r)}) [p(w_1^N | \hat{\mathbf{o}}_1^{T(r)}, \theta') \\ &\quad - p(w_1^N | s_1^{T(r)})] p(\hat{\mathbf{o}}_1^T, s_1^T | \theta') y(\hat{\mathbf{o}}_1^T, s_1^T, \theta', \theta) \end{aligned} \quad (\text{A.3.34})$$

$$\begin{aligned} \Delta_0(\theta) &= \sum_{s_1^T \in \mathcal{S}} \int_{\mathbf{o}_1^T} q_0(s_1^T) p(\mathbf{o}_1^T, s_1^T | \theta') y(\mathbf{o}_1^T, s_1^T, \theta', \theta) d\mathbf{o}_1^T \\ &= \sum_{s_1^T \in \mathcal{S}} d(s_1^T) \int_{\mathbf{o}_1^T} p(\mathbf{o}_1^T, s_1^T | \theta') y(\mathbf{o}_1^T, s_1^T, \theta', \theta) d\mathbf{o}_1^T \end{aligned} \quad (\text{A.3.35})$$

Since κ_ϵ (the subset of Θ for which $Q_\epsilon(\theta) > 0$) is a bounded set for sufficiently small ϵ (as shown in the proof of Theorem 3), it is contained within some compact set κ . So Theorem 4 is applied to bound the magnitude of $\Delta'(\theta)$ as shown in Equation A.3.36. The term $y_1(s_1^T, \theta', \theta)$ is given by Equation A.3.37.

$$\begin{aligned} &|\Delta'(\theta)| \\ &\leq \sum_{s_1^T \in \mathcal{S}} \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} L(w_1^N, \hat{w}_1^{M(r)}) \left| p(w_1^N | \hat{\mathbf{o}}_1^{T(r)}, \theta') - p(w_1^N | s_1^{T(r)}) \right| \left| p(\hat{\mathbf{o}}_1^T, s_1^T | \theta') y(\hat{\mathbf{o}}_1^T, s_1^T, \theta', \theta) \right| \\ &\leq \sum_{s_1^T \in \mathcal{S}} \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} L(w_1^N, \hat{w}_1^{M(r)}) \left| p(w_1^N | \hat{\mathbf{o}}_1^{T(r)}, \theta') - p(w_1^N | s_1^{T(r)}) \right| \frac{|y_1(s_1^T, \theta', \theta)|}{C(\hat{\mathbf{o}}_1^T, \theta', s_1^T)} \end{aligned} \quad (\text{A.3.36})$$

$$y_1(s_1^T, \theta', \theta) = \int_{\mathbf{o}_1^T} p(\mathbf{o}_1^T, s_1^T | \theta') y(\mathbf{o}_1^T, s_1^T, \theta', \theta) d\mathbf{o}_1^T \quad (\text{A.3.37})$$

The quantity $|\Delta_0(\theta)|$ is bounded below as shown in Equation A.3.38, where d_{min} is defined in Equation A.3.39.

$$\begin{aligned} |\Delta_0(\theta)| &= \left| \sum_{s_1^T \in \mathcal{S}} d(s_1^T) \int_{\mathbf{o}_1^T} p(\mathbf{o}_1^T, s_1^T | \theta') y(\mathbf{o}_1^T, s_1^T, \theta', \theta) d\mathbf{o}_1^T \right| \\ &\geq d_{min} \sum_{s_1^T \in \mathcal{S}} \left| \int_{\mathbf{o}_1^T} p(\mathbf{o}_1^T, s_1^T | \theta') y(\mathbf{o}_1^T, s_1^T, \theta', \theta) d\mathbf{o}_1^T \right| \end{aligned} \quad (\text{A.3.38})$$

$$d_{min} = \min_{s_1^T \in \mathcal{S}} \{d(s_1^T)\} \quad (\text{A.3.39})$$

From Equations A.3.36 and A.3.38, the inequality of Equation A.3.40 is derived, where L_{max} is defined in Equation A.3.41, C_{min} is defined in Equation A.3.42 and $|\mathcal{W}|$ is the number of members of the finite hypothesis space \mathcal{W} .

$$\begin{aligned} &|\Delta'(\theta)| \\ \leq &\sum_{s_1^T \in \mathcal{S}} \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} L(w_1^N, \hat{w}_1^{M(r)}) \left| p(w_1^N | \hat{\mathbf{o}}_1^{T(r)}, \theta') - p(w_1^N | s_1^{T(r)}) \right| \frac{|y_1(s_1^T, \theta', \theta)|}{C(\hat{\mathbf{o}}_1^T, \theta', s_1^T)} \\ \leq &\sum_{s_1^T \in \mathcal{S}} \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} \frac{L_{max}}{C_{min}} \left| \int_{\mathbf{o}_1^T} p(\mathbf{o}_1^T, s_1^T | \theta') y(\mathbf{o}_1^T, s_1^T, \theta', \theta) d\mathbf{o}_1^T \right| \\ \leq &\sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} \frac{L_{max}}{d_{min} C_{min}} |\Delta_0(\theta)| \\ = &\frac{R |\mathcal{W}| L_{max}}{d_{min} C_{min}} |\Delta_0(\theta)| \end{aligned} \quad (\text{A.3.40})$$

$$L_{max} = \max_{s_1^T \in \mathcal{S}, r, w_1^N \in \mathcal{W}} \{L(w_1^N, \hat{w}_1^{M(r)}) \left| \left[p(w_1^N | \hat{\mathbf{o}}_1^{T(r)}, \theta') - p(w_1^N | s_1^{T(r)}) \right] \right|\} \quad (\text{A.3.41})$$

$$C_{min} = \min_{s_1^T \in \mathcal{S}} \{C(\hat{\mathbf{o}}_1^T, \theta', s_1^T)\} \quad (\text{A.3.42})$$

Using Equations A.3.40 and A.3.33, and the fact that $\Delta_0(\theta)$ is non-negative, it can be demonstrated that $\Delta_\epsilon(\theta)$ is non-negative for $\theta \in \kappa_\epsilon$.

$$\begin{aligned} \Delta_\epsilon(\theta) &= \Delta_0(\theta) + \epsilon \Delta'(\theta) \\ &\geq \Delta_0(\theta) - \epsilon |\Delta'(\theta)| \\ &\geq \Delta_0(\theta) \left[1 - \epsilon \frac{R |\mathcal{W}| L_{max}}{d_{min} C_{min}} \right] \end{aligned} \quad (\text{A.3.43})$$

Provided $\epsilon \leq \frac{d_{\min} C_{\min}}{R \|\mathcal{W}\| L_{\max}}$, $\Delta_\epsilon(\theta)$ is non-negative for $\theta \in \kappa_\epsilon$. This proves Theorem 5.

Note that, without loss of generality, each $d(s_1^T)$ may be set to 1. In this case, provided $d'(s_1^T) \geq \frac{R \|\mathcal{W}\| L_{\max}}{C_{\min}}$, then Theorem 5 holds. This result will be used in the Section A.5 to obtain a lower bound for the learning rate in the extended Baum-Welch mean update formula. Firstly, the extended Baum-Welch mean update formula is derived from the previously-defined auxiliary function.

A.4 Extended Baum-Welch update formulae

In this section, the auxiliary function defined in Section A.2 is used to derive the extended Baum-Welch mean update equation. Using the notation of the previous section, Equation A.4.1 holds.

$$\theta_{\text{MBR}} \in \arg \min_{\theta} \int_{\mathbf{o}_1^T} \sum_{s_1^T \in \mathcal{S}} f(\mathbf{o}_1^T, s_1^T, \theta' | \theta') \log f(\mathbf{o}_1^T, s_1^T, \theta | \theta') d\mathbf{o}_1^T \quad (\text{A.4.1})$$

Differentiating with respect to some parameter x and setting the derivative to zero gives Equation A.4.2.

$$\int_{\mathbf{o}_1^T} \sum_{s_1^T \in \mathcal{S}} f(\mathbf{o}_1^T, s_1^T, \theta' | \theta') \nabla_x \log f(\mathbf{o}_1^T, s_1^T, \theta | \theta') d\mathbf{o}_1^T = 0 \quad (\text{A.4.2})$$

Note that Equation A.4.3 holds since only the factor $p(\mathbf{o}_1^T, s_1^T | \theta)$ in the definition of the function $f(\mathbf{o}_1^T, s_1^T, \theta | \theta')$ depends on θ .

$$\int_{\mathbf{o}_1^T} \sum_{s_1^T \in \mathcal{S}} f(\mathbf{o}_1^T, s_1^T, \theta' | \theta') \nabla_x (\log p(\mathbf{o}_1^T, s_1^T | \theta)) d\mathbf{o}_1^T = 0 \quad (\text{A.4.3})$$

Substituting Equation A.2.2 in Equation A.4.3 gives Equation A.4.4.

$$\begin{aligned} & \int_{\mathbf{o}_1^T} \sum_{s_1^T \in \mathcal{S}} p(\mathbf{o}_1^T, s_1^T | \theta') \left[\sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} 1_{\hat{\mathbf{o}}_1^T}(\mathbf{o}_1^T) a(w_1^N, \hat{w}_1^{M(r)}, \hat{\mathbf{o}}_1^{T(r)}, s_1^{T(r)}, \theta') 1_{\hat{\mathbf{o}}_1^T}(\mathbf{o}_1^T) \right. \\ & \quad \left. + d(s_1^T) \right] \nabla_x (\log p(\mathbf{o}_1^T, s_1^T | \theta)) d\mathbf{o}_1^T \\ & = 0 \end{aligned} \quad (\text{A.4.4})$$

For the sake of clarity the following definitions are used.

$$A = \int_{\mathbf{o}_1^T} \sum_{s_1^T} p(\mathbf{o}_1^T, s_1^T | \theta') \sum_{r=1}^R \sum_{w_1^N} 1_{\hat{\mathbf{o}}_1^T}(\mathbf{o}_1^T) L(w_1^N, \hat{w}_1^{M(r)}) p(w_1^N | s_1^{T(r)}) \nabla_x (\log p(\mathbf{o}_1^T, s_1^T | \theta)) d\mathbf{o}_1^T \quad (\text{A.4.5})$$

$$B = \int_{\mathbf{o}_1^T} \sum_{s_1^T} p(\mathbf{o}_1^T, s_1^T | \theta') \sum_{r=1}^R \sum_{w_1^N} 1_{\hat{\mathbf{o}}_1^T}(\mathbf{o}_1^T) L(w_1^N, \hat{w}_1^{M(r)}) p(w_1^N | \hat{\mathbf{o}}_1^T, \theta') \nabla_x (\log p(\mathbf{o}_1^T, s_1^T | \theta)) d\mathbf{o}_1^T \quad (\text{A.4.6})$$

$$C = \int_{\mathbf{o}_1^T} \sum_{s_1^T} p(\mathbf{o}_1^T, s_1^T | \theta') d(\hat{\mathbf{o}}_1^T, \theta') \nabla_x (\log p(\mathbf{o}_1^T, s_1^T | \theta)) d\mathbf{o}_1^T \quad (\text{A.4.7})$$

Using the above definitions Equation A.4.4 becomes Equation A.4.8.

$$B - A + C = 0 \quad (\text{A.4.8})$$

Equation A.4.5 can be rewritten as Equation A.4.9.

$$\begin{aligned} A &= \int_{\mathbf{o}_1^T} \sum_{s_1^T} p(\mathbf{o}_1^T, s_1^T | \theta') \sum_{r=1}^R \sum_{w_1^N} L(w_1^N, \hat{w}_1^{M(r)}) 1_{\hat{\mathbf{o}}_1^T}(\mathbf{o}_1^T) p(w_1^N | s_1^{T(r)}) \nabla_x (\log p(\mathbf{o}_1^T, s_1^T | \theta)) d\mathbf{o}_1^T \\ &= \sum_{s_1^T} p(\hat{\mathbf{o}}_1^T, s_1^T | \theta') \sum_{r=1}^R \sum_{w_1^N} L(w_1^N, \hat{w}_1^{M(r)}) p(w_1^N | s_1^{T(r)}) \nabla_x (\log p(\hat{\mathbf{o}}_1^T, s_1^T | \theta)) \\ &= p(\hat{\mathbf{o}}_1^T | \theta') \sum_{w_1^N} L(w_1^N, \hat{w}_1^{M(r)}) \prod_{k=1}^R \sum_{s_1^{T(k)}} p(s_1^{T(k)} | \hat{\mathbf{o}}_1^{T(k)}, \theta') \sum_{r=1}^R p(w_1^N | s_1^{T(r)}) \nabla_x (\log p(\hat{\mathbf{o}}_1^T, s_1^T | \theta)) \end{aligned} \quad (\text{A.4.9})$$

Similarly, Equation A.4.6 is rewritten as Equation A.4.10.

$$\begin{aligned} B &= \int_{\mathbf{o}_1^T} \sum_{s_1^T} p(\mathbf{o}_1^T, s_1^T | \theta') \sum_{r=1}^R \sum_{w_1^N} L(w_1^N, \hat{w}_1^{M(r)}) 1_{\hat{\mathbf{o}}_1^T}(\mathbf{o}_1^T) p(w_1^N | \hat{\mathbf{o}}_1^T, \theta') \nabla_x (\log p(\mathbf{o}_1^T, s_1^T | \theta)) d\mathbf{o}_1^T \\ &= \sum_{s_1^T \in \mathcal{S}} p(\hat{\mathbf{o}}_1^T, s_1^T | \theta') \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} L(w_1^N, \hat{w}_1^{M(r)}) p(w_1^N | \hat{\mathbf{o}}_1^T, \theta') \nabla_x (\log p(\hat{\mathbf{o}}_1^T, s_1^T | \theta)) \\ &= p(\hat{\mathbf{o}}_1^T | \theta') \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} L(w_1^N, \hat{w}_1^{M(r)}) p(w_1^N | \hat{\mathbf{o}}_1^T, \theta') \sum_{s_1^T \in \mathcal{S}} p(s_1^T | \hat{\mathbf{o}}_1^T, \theta') \nabla_x (\log p(\hat{\mathbf{o}}_1^T, s_1^T | \theta)) \end{aligned} \quad (\text{A.4.10})$$

The mean update equation is now derived for the case of first order HMMs with Gaussian output distributions.

A.4.1 Means of Gaussian state output distributions

Let $\boldsymbol{\mu}_s$ be the mean of the Gaussian output distribution of HMM state s . Since the state sequence probability $p(s_1^T | \theta)$ is independent of $\boldsymbol{\mu}_s$ (it depends only on state transition probabilities) the following equation holds.

$$\nabla_{\boldsymbol{\mu}_s} (\log p(\hat{\mathbf{o}}_1^{T(k)}, s_1^{T(k)} | \theta)) = \nabla_{\boldsymbol{\mu}_s} (\log p(\hat{\mathbf{o}}_1^{T(k)} | s_1^{T(k)}, \theta)) \quad (\text{A.4.11})$$

Equation A.4.12 expresses the probability of a sequence of observations, conditioned on a state sequence.

$$p(\hat{\mathbf{o}}_1^{T(k)} | s_1^{T(k)}, \theta) = \prod_{t=1}^{T(k)} p(\hat{\mathbf{o}}_t(k) | s_t(k), \theta) \quad (\text{A.4.12})$$

Here $\hat{\mathbf{o}}_t(k)$ is the t -th element of the training set observation sequence $\hat{\mathbf{o}}_1^{T(k)}$, $T(k)$ is the length of this observation sequence and $s_t(k)$ is the t -th element of the state sequence $s_1^{T(k)}$. Equation A.4.13 holds, where \mathbf{C}_s is the covariance matrix of state s .

$$\nabla_{\boldsymbol{\mu}_s} (\log p(\hat{\mathbf{o}}_t | s, \boldsymbol{\mu}_s)) = \mathbf{C}_s^{-1} (\hat{\mathbf{o}}_t - \boldsymbol{\mu}_s) \quad (\text{A.4.13})$$

So Equation A.4.11 may be re-written as Equation A.4.14.

$$\nabla_{\boldsymbol{\mu}_s} (\log p(\hat{\mathbf{o}}_1^{T(k)}, s_1^{T(k)} | \theta)) = \sum_{t=1}^{T(k)} 1_s(s_t(k)) \mathbf{C}_s^{-1} (\hat{\mathbf{o}}_t(k) - \boldsymbol{\mu}_s) \quad (\text{A.4.14})$$

Equation A.4.14 is extended to Equation A.4.15.

$$\nabla_{\boldsymbol{\mu}_s} (\log p(\hat{\mathbf{o}}_1^T, s_1^T | \theta)) = \sum_{k=1}^R \sum_{t=1}^{T(k)} 1_s(s_t(k)) \mathbf{C}_s^{-1} (\hat{\mathbf{o}}_t(k) - \boldsymbol{\mu}_s) \quad (\text{A.4.15})$$

Setting the parameter x to $\boldsymbol{\mu}_s$ and expanding the summation over the state sequences in Equation A.4.10 yields Equation A.4.16.

$$\begin{aligned} & \sum_{s_1^T \in \mathcal{S}} p(s_1^T | \hat{\mathbf{o}}_1^T, \theta') \nabla_{\boldsymbol{\mu}_s} (\log p(\hat{\mathbf{o}}_1^T, s_1^T | \theta)) \\ &= \sum_{s_1^T \in \mathcal{S}} p(s_1^T | \hat{\mathbf{o}}_1^T, \theta') \sum_{j=1}^R \sum_{t=1}^{T(j)} [1_s(s_t(j)) \mathbf{C}_s^{-1} (\hat{\mathbf{o}}_t(j) - \boldsymbol{\mu}_s)] \\ &= \prod_{k=1}^R \sum_{s_1^{T(k)} \in \mathcal{S}_k} p(s_1^{T(k)} | \hat{\mathbf{o}}_1^{T(k)}, \theta') \sum_{j=1}^R \sum_{t=1}^{T(j)} [1_s(s_t(j)) \mathbf{C}_s^{-1} (\hat{\mathbf{o}}_t(j) - \boldsymbol{\mu}_s)] \quad (\text{A.4.16}) \end{aligned}$$

Using Equation A.2.31 with $\mathbf{g}(\hat{\mathbf{o}}_t(j), s)$ equal to $\mathbf{C}_s^{-1} (\hat{\mathbf{o}}_t(j) - \boldsymbol{\mu}_s)$, Equation A.4.16 is rephrased as Equation A.4.17.

$$\begin{aligned} \sum_{s_1^T \in \mathcal{S}} p(s_1^T | \hat{\mathbf{o}}_1^T, \theta') \nabla_{\boldsymbol{\mu}_s} (\log p(\hat{\mathbf{o}}_1^T, s_1^T | \theta)) &= \sum_{j=1}^R \sum_{t=1}^{T(j)} \gamma_s(t, j) \mathbf{C}_s^{-1} (\hat{\mathbf{o}}_t(j) - \boldsymbol{\mu}_s) \\ &= \mathbf{C}_s^{-1} \sum_{j=1}^R [\boldsymbol{\theta}_s(j) - \gamma_s(j) \boldsymbol{\mu}_s] \quad (\text{A.4.17}) \end{aligned}$$

Above $\gamma_s(j)$ and $\theta_s(j)$ are defined in Equations A.4.18 and A.4.19 respectively.

$$\gamma_s(j) = \sum_{s_1^{T(j)} \in \mathcal{S}_j} p(s_1^{T(j)} | \hat{\mathbf{o}}_1^{T(j)}, \theta') \sum_{t=1}^{T(j)} 1_s(s_t(j)) \quad (\text{A.4.18})$$

$$\theta_s(j) = \sum_{s_1^{T(j)} \in \mathcal{S}_j} p(s_1^{T(j)} | \hat{\mathbf{o}}_1^{T(j)}, \theta') \sum_{t=1}^{T(j)} 1_s(s_t(j)) \mathbf{o}_t(j) \quad (\text{A.4.19})$$

Setting x to μ_s and expanding the summation over the state sequences in Equation A.4.9 yields Equation A.4.20.

$$\begin{aligned} & \prod_{k=1}^R \sum_{s_1^{T(k)} \in \mathcal{S}_k} p(s_1^{T(k)} | \hat{\mathbf{o}}_1^{T(k)}, \theta') \sum_{r=1}^R p(w_1^N | s_1^{T(r)}) \nabla_{\mu_s} (\log p(\hat{\mathbf{o}}_1^T, s_1^T | \theta)) \\ &= \prod_{k=1}^R \sum_{s_1^{T(k)} \in \mathcal{S}_k} p(s_1^{T(k)} | \hat{\mathbf{o}}_1^{T(k)}, \theta') \sum_{r=1}^R p(w_1^N | s_1^{T(r)}) \sum_{t=1}^T [1_s(s_t) \mathbf{C}_s^{-1}(\hat{\mathbf{o}}_t - \mu_s)] \quad (\text{A.4.20}) \end{aligned}$$

Using Equation A.2.37 with $\mathbf{g}(\hat{\mathbf{o}}_t, s_t) = [1_s(s_t) \mathbf{C}_s^{-1}(\hat{\mathbf{o}}_t - \mu_s)]$ in Equation A.4.20 gives Equation A.4.21, where the outer sum is over all states s' .

$$\begin{aligned} & \prod_{k=1}^R \sum_{s_1^{T(k)} \in \mathcal{S}_k} p(s_1^{T(k)} | \hat{\mathbf{o}}_1^{T(k)}, \theta') \sum_{r=1}^R p(w_1^N | s_1^{T(r)}) \nabla_{\mu_s} (\log p(\hat{\mathbf{o}}_1^T, s_1^T | \theta)) \\ &= p(w_1^N | \hat{\mathbf{o}}_1^{T(r)}, \theta') \sum_{s'} 1_{s'}(s) \mathbf{C}_s^{-1} \left[\sum_{j=1, j \neq r}^R \sum_{t=1}^{T(j)} \gamma_s(t, j) [(\hat{\mathbf{o}}_t(j) - \mu_s)] \right. \\ & \quad \left. + \sum_{t=1}^{T(r)} \gamma_s(t, r, w_1^N) [(\hat{\mathbf{o}}_t(j) - \mu_s)] \right] \\ &= p(w_1^N | \hat{\mathbf{o}}_1^{T(r)}, \theta') \mathbf{C}_s^{-1} \left[\sum_{j=1, j \neq r}^R [\theta_s(j) - \gamma_s(j) \mu_s] + [\theta_s(r, w_1^N) - \gamma_s(r, w_1^N) \mu_s] \right] \quad (\text{A.4.21}) \end{aligned}$$

The quantities $\gamma_s(r, w_1^N)$ and $\theta_s(r, w_1^N)$ are defined in Equations A.4.22 and A.4.23 respectively.

$$\gamma_s(r, w_1^N) = \sum_{s_1^{T(r)} \in \mathcal{S}_r} p(s_1^{T(r)} | w_1^N, \hat{\mathbf{o}}_1^{T(r)}, \theta') \sum_{t=1}^{T(r)} 1_s(s_t^t) \quad (\text{A.4.22})$$

$$\theta_s(r, w_1^N) = \sum_{s_1^{T(r)} \in \mathcal{S}_k} p(s_1^{T(r)} | w_1^N, \hat{\mathbf{o}}_1^{T(k)}, \theta') \sum_{t=1}^{T(r)} 1_s(s_r^t) \hat{\mathbf{o}}_t(r) \quad (\text{A.4.23})$$

Substituting Equation A.4.17 into Equation A.4.10 gives Equation A.4.24.

$$\begin{aligned}
\mathbf{B} &= p(\hat{\mathbf{o}}_1^T | \theta') \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} L(w_1^N, \hat{w}_1^{M(r)}) p(w_1^N | \hat{\mathbf{o}}_1^{T(r)}, \theta') \sum_{s_1^T \in \mathcal{S}} p(s_1^T | \hat{\mathbf{o}}_1^T, \theta') \nabla_x (\log p(\hat{\mathbf{o}}_1^T, s_1^T | \theta)) \\
&= p(\hat{\mathbf{o}}_1^T | \theta') \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} L(w_1^N, \hat{w}_1^{M(r)}) p(w_1^N | \hat{\mathbf{o}}_1^{T(r)}, \theta') \mathbf{C}_s^{-1} \sum_{k=1}^R [\boldsymbol{\theta}_s(k) - \gamma_s(k) \boldsymbol{\mu}_s] \quad (\text{A.4.24})
\end{aligned}$$

Substituting Equation A.4.21 into Equation A.4.9 gives Equation A.4.25.

$$\begin{aligned}
\mathbf{A} &= p(\hat{\mathbf{o}}_1^T | \theta) \sum_{w_1^N} L(w_1^N, \hat{w}_1^{M(r)}) \prod_{k=1}^R \sum_{s_1^{T(k)}} p(s_1^{T(k)} | \hat{\mathbf{o}}_1^{T(k)}, \theta') \sum_{r=1}^R p(w_1^N | s_1^{T(r)}) \nabla_x (\log p(\hat{\mathbf{o}}_1^T, s_1^T | \theta)) \\
&= p(\hat{\mathbf{o}}_1^T | \theta) \sum_{w_1^N \in \mathcal{W}} L(w_1^N, \hat{w}_1^{M(r)}) p(w_1^N | \hat{\mathbf{o}}_1^{T(r)}, \theta') \mathbf{C}_s^{-1} \left[\sum_{j=1, j \neq r}^R [\boldsymbol{\theta}_s(j) - \gamma_s(j) \boldsymbol{\mu}_s] \right. \\
&\quad \left. + [\boldsymbol{\theta}_s(r, w_1^N) - \gamma_s(r, w_1^N) \boldsymbol{\mu}_s] \right] \quad (\text{A.4.25})
\end{aligned}$$

Equation A.4.7 may be rewritten as Equation A.4.26.

$$\begin{aligned}
\mathbf{C} &= \int_{\mathbf{o}_1^T} \sum_{s_1^T \in \mathcal{S}} p(\mathbf{o}_1^T, s_1^T | \theta') d(s_1^T) \nabla_x (\log p(\mathbf{o}_1^T, s_1^T | \theta)) d\mathbf{o}_1^T \\
&= \int_{\mathbf{o}_1^T} \sum_{s_1^T \in \mathcal{S}} p(\mathbf{o}_1^T, s_1^T | \theta') d(s_1^T) \sum_{k=1}^R \sum_{t=1}^{T(k)} 1_s(s_t(k)) \mathbf{C}_s^{-1} (\mathbf{o}_t(k) - \boldsymbol{\mu}_s) d\mathbf{o}_1^T \quad (\text{A.4.26})
\end{aligned}$$

The current mean of state s , $\hat{\boldsymbol{\mu}}_s$, may be expressed as Equation A.4.27.

$$\hat{\boldsymbol{\mu}}_s = \frac{\int_{\mathbf{o}_1^T} \sum_{s_1^T \in \mathcal{S}} d(s_1^T) \sum_{k=1}^R \sum_{t=1}^{T(k)} 1_s(s_t(k)) p(\mathbf{o}_1^T, s_1^T | \theta') \mathbf{o}_t(k) d\mathbf{o}_1^T}{\int_{\mathbf{o}_1^T} \sum_{s_1^T \in \mathcal{S}} d(s_1^T) \sum_{k=1}^R \sum_{t=1}^{T(k)} 1_s(s_t(k)) p(\mathbf{o}_1^T, s_1^T | \theta') d\mathbf{o}_1^T} \quad (\text{A.4.27})$$

Note that D_s (see Equation A.2.16) may be re-expressed as in Equation A.4.28.

$$D_s = \int_{\mathbf{o}_1^T} \sum_{s_1^T \in \mathcal{S}} d(s_1^T) \sum_{k=1}^R \sum_{t=1}^{T(k)} 1_s(s_t(k)) p(\mathbf{o}_1^T, s_1^T | \theta') d\mathbf{o}_1^T \quad (\text{A.4.28})$$

Substituting Equations A.4.27 and A.4.28 in Equation A.4.26 gives Equation A.4.29.

$$\mathbf{C} = D_s \mathbf{C}_s^{-1} (\hat{\boldsymbol{\mu}}_s - \boldsymbol{\mu}_s) \quad (\text{A.4.29})$$

Using Equations A.4.24 and A.4.25 gives Equation A.4.30.

$$\begin{aligned}
& \mathbf{B} - \mathbf{A} \\
&= p(\hat{\boldsymbol{\theta}}_1^T | \theta') \mathbf{C}_s^{-1} \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} L(w_1^N, \hat{w}_1^{M(r)}) p(w_1^N | \hat{\boldsymbol{\theta}}_1^{T(r)}, \theta') \left[\gamma_s(r, w_1^N) \boldsymbol{\mu}_s \right. \\
&\quad \left. - \boldsymbol{\theta}_s(r, w_1^N) - [\gamma_s(r) \boldsymbol{\mu}_s - \boldsymbol{\theta}_s(r)] \right] \quad (\text{A.4.30})
\end{aligned}$$

Using Equations A.4.8, A.4.29 and A.4.30, premultiplying by \mathbf{C}_s and dividing by $p(\hat{\boldsymbol{\theta}}_1^T | \theta')$ gives Equation A.4.31.

$$\begin{aligned}
\mathbf{0} &= \mathbf{B} - \mathbf{A} + \mathbf{C} \\
&= \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} L(w_1^N, \hat{w}_1^{M(r)}) p(w_1^N | \hat{\boldsymbol{\theta}}_1^{T(r)}, \theta') [\boldsymbol{\theta}_s(r) - \boldsymbol{\theta}_s(r, w_1^N) - (\gamma_s(r) - \gamma_s(r, w_1^N)) \boldsymbol{\mu}_s] \\
&\quad + \frac{D_s}{p(\hat{\boldsymbol{\theta}}_1^T | \theta')} (\hat{\boldsymbol{\mu}}_s - \boldsymbol{\mu}_s) \quad (\text{A.4.31})
\end{aligned}$$

Rearranging Equation A.4.31 gives Equation A.4.32.

$$\begin{aligned}
& \boldsymbol{\mu}_s \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} L(w_1^N, \hat{w}_1^{M(r)}) p(w_1^N | \hat{\boldsymbol{\theta}}_1^{T(r)}, \theta') (\gamma_s(r) - \gamma_s(r, w_1^N)) \\
&= \sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} L(w_1^N, \hat{w}_1^{M(r)}) p(w_1^N | \hat{\boldsymbol{\theta}}_1^{T(r)}, \theta') [\boldsymbol{\theta}_s(r) - \boldsymbol{\theta}_s(r, w_1^N)] + D'_s (\hat{\boldsymbol{\mu}}_s - \boldsymbol{\mu}_s) \quad (\text{A.4.32})
\end{aligned}$$

The term D'_s is defined in Equation A.4.33.

$$D'_s = \frac{D_s}{p(\hat{\boldsymbol{\theta}}_1^T | \theta')} \quad (\text{A.4.33})$$

Notice that $\sum_{w_1^N \in \mathcal{W}} L(w_1^N, \hat{w}_1^{M(r)}) p(w_1^N | \hat{\boldsymbol{\theta}}_1^{T(r)}, \theta') \gamma_s(r)$ may be re-expressed as shown in Equation A.4.34.

$$\begin{aligned}
& \sum_{w_1^N \in \mathcal{W}} L(w_1^N, \hat{w}_1^{M(r)}) p(w_1^N | \hat{\boldsymbol{\theta}}_1^{T(r)}, \theta') \gamma_s(r) \\
&= \gamma_s(r) \sum_{w_1^N \in \mathcal{W}} \left[\gamma_s(r, w_1^N) \frac{\sum_{w_1^N \in \mathcal{W}} L(w_1^N, \hat{w}_1^{M(r)}) p(w_1^N | \hat{\boldsymbol{\theta}}_1^{T(r)}, \theta')}{\sum_{w_1^N \in \mathcal{W}} \gamma_s(r, w_1^N)} \right] \quad (\text{A.4.34})
\end{aligned}$$

Letting $l_{av}^r = \sum_{w_1^N \in \mathcal{W}} L(w_1^N, \hat{w}_1^{M(r)}) p(w_1^N | \hat{\mathbf{o}}_1^{T(r)}, \theta')$, Equation A.4.34 becomes Equation A.4.35.

$$\begin{aligned} \sum_{w_1^N \in \mathcal{W}} L(w_1^N, \hat{w}_1^{M(r)}) p(w_1^N | \hat{\mathbf{o}}_1^{T(r)}, \theta') \gamma_s(r) &= l_{av}^r \gamma_s(r) \sum_{w_1^N \in \mathcal{W}} \frac{\gamma_s(r, w_1^N)}{\sum_{w_1^N \in \mathcal{W}} \gamma_s(r, w_1^N)} \\ &= l_{av}^r \gamma_s(r) \end{aligned} \quad (\text{A.4.35})$$

Notice further that $\gamma_s(r)$ can be expressed as in Equation A.4.36.

$$\gamma_s(r) = \sum_{w_1^N \in \mathcal{W}} P(w_1^N | \hat{\mathbf{o}}_1^{T(r)}, \theta') \gamma_s(r, w_1^N) \quad (\text{A.4.36})$$

Using Equations A.4.35 and A.4.36 in Equation A.4.32 gives Equation A.4.37.

$$\sum_{w_1^N \in \mathcal{W}} L(w_1^N, \hat{w}_1^{M(r)}) p(w_1^N | \hat{\mathbf{o}}_1^{T(r)}, \theta') \gamma_s(r) = \sum_{w_1^N \in \mathcal{W}} p(w_1^N | \hat{\mathbf{o}}_1^{T(r)}, \theta') \gamma_s(r, w_1^N) l_{av}^r \quad (\text{A.4.37})$$

Similar rearrangement yields Equation A.4.38.

$$\sum_{w_1^N \in \mathcal{W}} L(w_1^N, \hat{w}_1^{M(r)}) p(w_1^N | \hat{\mathbf{o}}_1^{T(r)}, \theta') \boldsymbol{\theta}_s(r) = \sum_{w_1^N \in \mathcal{W}} p(w_1^N | \hat{\mathbf{o}}_1^{T(r)}, \theta') \boldsymbol{\theta}_s(r, w_1^N) l_{av}^r \quad (\text{A.4.38})$$

Substituting Equations A.4.37 and A.4.38 in Equation A.4.32 and rearranging yields the extended Baum-Welch mean update equation, Equation A.4.39.

$$\boldsymbol{\mu}_s = \frac{\sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} p(w_1^N | \hat{\mathbf{o}}_1^{T(r)}, \theta') \boldsymbol{\theta}_s(r, w_1^N) \left[l_{av}^r - L(w_1^N, \hat{w}_1^{M(r)}) \right] + D'_s \hat{\boldsymbol{\mu}}_s}{\sum_{r=1}^R \sum_{w_1^N \in \mathcal{W}} p(w_1^N | \hat{\mathbf{o}}_1^{T(r)}, \theta') \gamma_s(r, w_1^N) \left[l_{av}^r - L(w_1^N, \hat{w}_1^{M(r)}) \right] + D'_s} \quad (\text{A.4.39})$$

A.5 Lower bound on learning rate D

The learning rate term D'_s of Equation A.4.39 may be bounded below. Equations A.4.33 and A.2.16 yield Equation A.5.1.

$$D'_s = \frac{1}{p(\hat{\mathbf{o}}_1^T | \theta')} \sum_{s_1^T \in \mathcal{S}} d(s_1^T) \sum_{t=1}^T 1_s(s_t) p(s_1^T) \quad (\text{A.5.1})$$

Recall from Section A.3 that provided $d(s_1^T) \geq \frac{R \|\mathcal{W}\| L_{max}}{C_{min}}$ then the defined function is a valid auxiliary function for the MBR criterion. Using this result, a lower bound on the learning rate term D'_s is derived as shown in Equation A.5.2.

$$D'_s \geq \frac{R \|\mathcal{W}\| L_{max}}{C_{min} p(\hat{\mathbf{o}}_1^T | \theta')} \sum_{s_1^T \in \mathcal{S}} \sum_{t=1}^T 1_s(s_t) p(s_1^T) \quad (\text{A.5.2})$$

Appendix B

Experimental Systems

B.1 Baseline system

The systems used for the experimental work of this thesis are based upon the individual headset microphone (IHM) 2005 AMI meeting speech transcription system (Hain et al. (2005a), Hain et al. (2005b)). This section gives further information on these systems including details on acoustic features, acoustic and language models, system operation and datasets used in training and evaluation.

B.1.1 Acoustic features

The IHM system uses 39-dimensional features to represent the speech signal. In the first pass this feature vector comprises 13 Mel-frequency-based perceptual linear prediction coefficients (MF-PLP, Woodland et al. (1997)) as well as the first and second time derivatives of these features.

In subsequent passes the feature vector is extracted via smoothed heteroscedastic linear discriminant analysis (SHLDA, Burget (2004)) from a 52-dimensional vector comprising 13 MF-PLP coefficients and the first, second and third time derivatives of these features. Additionally, in the second pass, these features are normalised using speaker-specific vocal tract length normalisation (Lee and Rose (1996)) as well as speaker-specific cepstral mean (CMN) and variance (CVN) normalisation (Atal (1974)).

B.1.2 Acoustic models

The acoustic models are triphone HMMs with three emitting states and left-to-right topology. The model states are clustered using a phonetic decision tree (Young et al. (1994)) and trained using 104 hours of speech (see Section B.1.5) and the maximum likelihood criterion. Approximately 4000 tied states are used and state output distributions are modelled with 16-component Gaussian mixture models.

B.1.3 Dictionary and language models

The recognition dictionary contains the 50000 most frequently used words as specified by a procedure outlined in Hain et al. (2005b). The pronunciations are based on the UNISYN pronunciation lexicon (Fitt (2000)).

Language models derived from several text corpora are estimated using ML-based techniques. A trigram language model suitable for meeting speech is then interpolated from these language models. More details of this procedure and the text corpora are found in Hain et al. (2005a) and Hain et al. (2005b).

B.1.4 System operation

As illustrated in Figure B.1, three passes¹ are used for recognition. All passes use the same trigram language model, details of which are found in (Hain et al. (2005a)). The ML-estimated acoustic models are used in the first recognition pass, the output of which is used for unsupervised estimation of speaker-specific VTLN, CMN and CVN normalisation parameters. A language model scaling factor of 14.0 is used in recognition.

The second recognition pass uses ML-estimated acoustic models which incorporate a smoothed HLDA transform of the acoustic features. Cepstral mean and variance normalisation as well as VTLN are used to normalise the features. The estimated transcription given by the first recognition pass is used to calculate the parameters for these feature normalisation techniques. A language model scaling factor of 14.0 is used in recognition.

A third pass is used only in those experiments using acoustic model adaptation. Details of the adaptation procedures and the third recognition pass are detailed in Section 8.5.

B.1.5 Training and evaluation datasets

The training and evaluation datasets used to assess the techniques presented in this thesis are recordings of spontaneous speech in meetings. Each meeting participants wore a head-mounted microphone during recording. While the language of all meetings is English, not all speakers are native English speakers.

The training dataset used to estimate acoustic models comprised 104 hours of transcribed speech in meetings from a selection of corpora including the ICSI meeting corpus (Janin et al. (2003)), the NIST meeting room pilot corpus (Garofolo et al. (2004)), the ISL meeting corpus (Burger et al. (2002)), the NIST RT04s development and evaluation sets (*rt04sdev* and *rt04seval*) and the AMI meeting corpus (Carletta et al. (2005)).

The National Institute of Standards and Technology (NIST) conference meeting evaluation datasets are used as test data². These datasets are labelled *rt05seval*, *rt06seval* and *rt07seval*. The number of hours of speech, speakers and words (including hesitations, partial words and fillers) associated with each dataset is shown in Table B.1.

¹Note that the full IHM 2005 system uses six recognition passes. A simplified system is used in the experimental work of this thesis.

²More details of these datasets are found at <http://www.nist.gov/speech/tests/rt/>.

Dataset	<i>rt05seval</i>	<i>rt06seval</i>	<i>rt07seval</i>
# hours	1.9	2.4	3.1
# speakers	53	43	31
# words	24776	33321	37314

Table B.1: *NIST conference meeting speech evaluation datasets.*

B.2 MBR-estimated system operation

The system used to evaluate MBR-estimated acoustic models (Sections 6.5 and 7.7) is based upon the baseline system described in Section B.1. Only the first two recognition passes are used. The first recognition pass is identical to the first pass described in Section B.1. The second pass is also identical to the second pass described in Section B.1, with the exception that the MBR-estimated acoustic models replace the ML-estimated models.

B.3 MBR parameter estimation details

The details of MBR parameter re-estimation are illustrated in Figure B.2. The ML-estimated models used in the second recognition pass of the baseline system (Section B.1) are used as a starting point for MBR re-estimation. Using these models, a lattice is generated for each training set utterance using a recognition pass and a bigram language model. The bigram language model is derived from the trigram used in recognition. A language model scaling factor of 13.0 and an insertion penalty of -10.0 are used at this stage. These lattices are referred to as the denominator lattices.

The same ML models are again used in a recognition pass to generate a lattice comprising the most likely alignments of the correct transcript for each training-set utterance. These lattices are referred to as the numerator lattices. The most probable alignment is chosen as the reference alignment in MBR parameter estimation. Note that all lattices are phoneme, model and state-marked to accommodate the optimisation of the sub-word MBR criteria. This means that each arc (which represents an aligned word) has associated time-aligned sequences corresponding to the most likely phoneme, model and state alignments of the aligned word. The lattice can thus be considered as a sample of alignments corresponding to word and sub-word hypothesis spaces.

The numerator and denominator lattices are then rescored with a unigram language model (again, derived from the trigram used in recognition) to improve the MBR-estimated model generalisation as explained in Section 5.2.5. The denominator lattices are subsequently pruned to a maximum density of 200.0 arcs per second to reduce the computational cost of the MBR criterion optimisation process. The alignments present in the unigram numerator lattices are then added into the corresponding unigram denominator lattices to yield the final lattices used in MBR parameter re-estimation.

When gathering the statistics used for parameter estimation, the probability distribution described by the unigram LM is further broadened by using an LM scaling factor of 0.5. An

acoustic scaling factor of $\frac{1}{14}$ is used, the inverse of the LM scaling factor used in the second recognition pass. The occupancy-dependent scheme for setting the learning rate described in Section 5.2.3 is used, and the constant E is set to 0.5. The I-smoothing technique (Section 5.2.4) is not used unless specified otherwise.

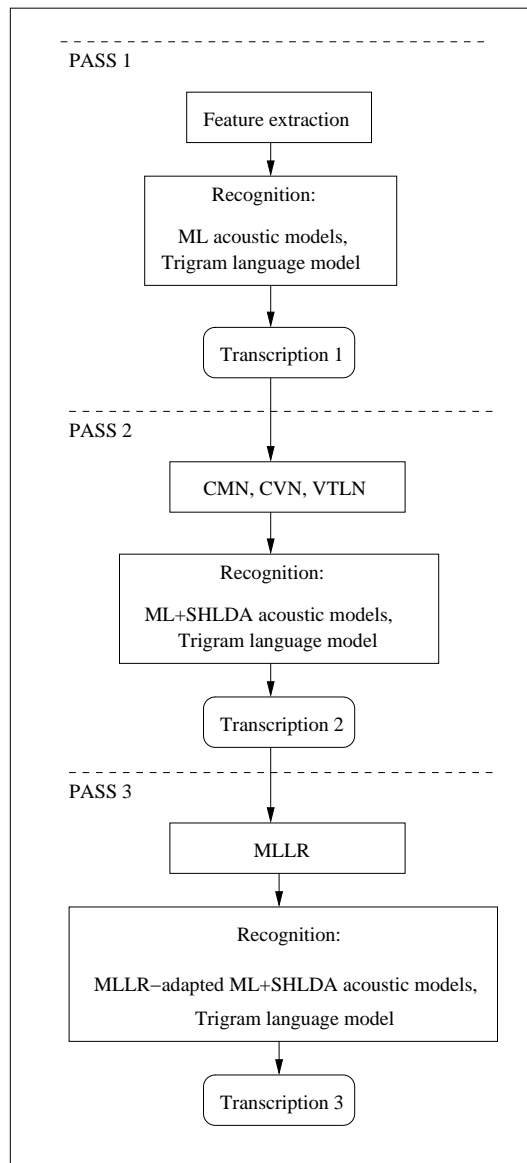
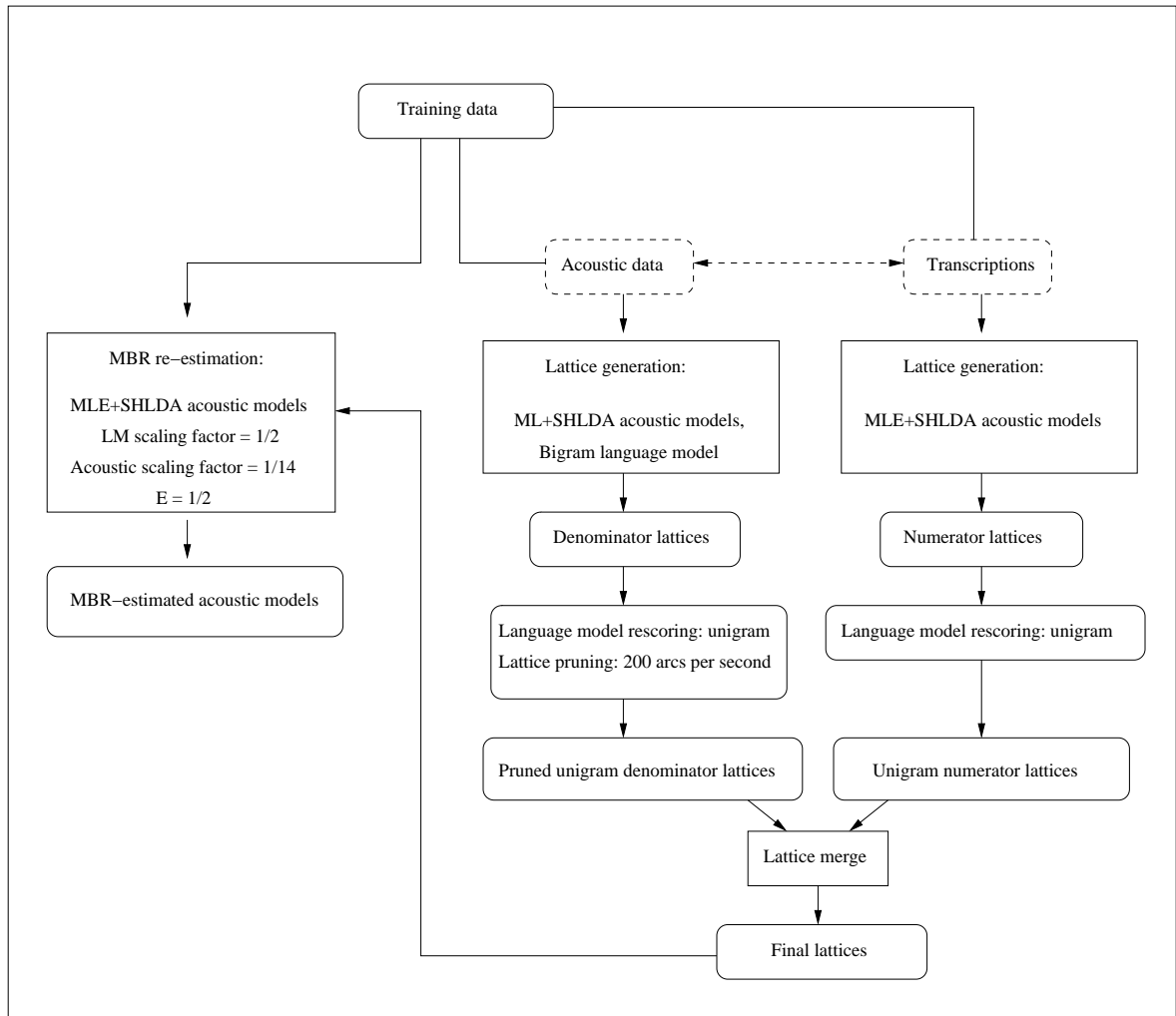


Figure B.1: 2005 AMI meeting speech transcription system.

Figure B.2: *Details of MBR parameter re-estimation.*

Bibliography

- Ahadi, S., Woodland, P. (1997). Combined Bayesian and predictive techniques for rapid speaker adaptation of continuous density HMMs. *Computer Speech and Language* 11 pp. 187–206.
- Anastasakos, T., McDonough, J., Schwartz, R., Makhoul, J. (1996). A compact model for speaker adaptive training. *Proceedings ICSLP*.
- Atal, B. S. (1974). Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification. *Journal of the Acoustical Society of America* 6 pp. 1304–1312.
- Attias, H. (2000). A variational Bayesian framework for graphical models. *Advances in Neural Information Processing Systems 12*.
- Axelrod, S., Goel, V., Gopinath, R., Olsen, P., Visweswariah, K. (2007). Discriminative estimation of subspace constrained Gaussian mixture models for speech recognition. *IEEE Transactions on Audio, Speech, and Language Processing* 15(1) pp. 172–189.
- Baum, L., Petrie, T., Soules, G., Weiss, N. (1970). A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *Annals of Mathematical Statistics*.
- Bogert, B. P., Healy, M. J. R., Tukey, J. W. (1963). The quefrency analysis of time series for echoes: cepstrum, pseudo-autocovariance, cross-cepstrum, and saphe cracking. *Proceedings Symposium on Time Series Analysis*.
- Bridle, J. S. (1989). *Neurocomputing: algorithms, architectures and applications* (Probabilistic interpretation of feedforward classification network outputs with relationships to statistical pattern recognition). Springer-Verlag.
- Bub, T., Schwinn, J. (1996). VERBMOBIL: The evolution of a complex large speech-to-speech translation system. *Proceedings ICSLP*.
- Burger, S., MacLaren, V., Yu, H. (2002). The ISL meeting corpus: The impact of meeting type on speech style. *Proceedings Interspeech*.

- Burget, L. (2004). Combination of speech features using smoothed heteroscedastic linear discriminant analysis. *Proceedings Interspeech*.
- Carletta, J., Ashby, S., Bourban, S., Guillemot, M., Kronenthal, M., Lathoud, G., Lincoln, M., McCowan, I., Hain, T., Kraaij, W., Post, W., Kadlec, J., Wellner, P., Flynn, M., Reidsma, D. (2005). The AMI meeting corpus. *Proceedings MLMI*.
- Cole, R., Muthusamy, Y., Fauty, M. (1990). The ISOLET spoken letter database. Technical report, Oregon Graduate Institute.
- Cox, S. (1995). Predictive speaker adaptation in speech recognition. *Computer Speech and Language* 9 pp. 1–17.
- Cox, S., Dasmahapatra, S. (2002). High-level approaches to confidence estimation in speech recognition. *IEEE Transactions on Speech and Audio Processing* 10(7) pp. 460–471.
- Davis, S., Mermelstein, P. (1980). Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Transactions on Acoustics, Speech, and Signal Processing* 28(4) pp. 357–366.
- Dempster, A., Laird, N., Rubin, D. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B* 39(1) pp. 1–38.
- Doumpiotis, V., Byrne, W. (2004). Pinched lattice minimum Bayes risk discriminative training for large vocabulary continuous speech recognition. *Proceedings Interspeech*.
- Doumpiotis, V., Byrne, W. (2005). Lattice segmentation and minimum Bayes risk discriminative training for large vocabulary continuous speech recognition. *Speech Communication* 48(2) pp. 142–160.
- Du, J., Liu, P., Soong, F., Zhou, J., Wang, R. (2006). Minimum divergence based discriminative training. *Proceedings Interspeech*.
- Eisele, T., Haeb-umbach, R., Langmann, D. (1996). A comparative study of linear feature transformation techniques for automatic speech recognition. *Proceedings ICASSP*.
- Evermann, G., Woodland, P. (2000). Large vocabulary decoding and confidence estimation using word posterior probabilities. *Proceedings ICASSP*.
- Finke, M., Zeppenfeld, T., Maier, M., Mayfield, L., Ries, K., Zhan, P., Lafferty, J., Waibel, A. (1996). Switchboard April 1996 evaluation report. *Proceedings LVCSR Hub 5 Workshop*.
- Fitt, S. (2000). Documentation and user guide to UNISYN lexicon and post-lexical rules. Technical report, Centre for Speech Technology Research, Edinburgh University.
- Gales, M., Woodland, P. (1996). Mean and variance adaptation within the MLLR framework. *Computer Speech and Language* 10(4) pp. 249–264.

- Gales, M. J. F. (1996). The generation and use of regression class trees for MLLR adaptation. Technical report, Engineering Department, Cambridge University.
- Gales, M. J. F. (2000). Cluster adaptive training of hidden Markov models. *IEEE Transactions on Speech and Audio Processing* 8 pp. 417–428.
- Garofolo, J., Laprun, C., Michel, M., Stanford, V., Tabassi, E. (2004). The NIST meeting room pilot corpus. *Proceedings Language Resources and Evaluation (LREC)*.
- Gauvain, J., Lee, C. (1994). Maximum a-posteriori estimation for multivariate Gaussian mixture observations of Markov chains. *IEEE Transactions on Speech and Audio Processing* 2 pp. 291–298.
- Gibson, M., Hain, T. (2007). Temporal masking for unsupervised minimum Bayes risk speaker adaptation. *Proceedings Interspeech*.
- Gillick, L., Cox, S. (1989). Some statistical issues in the comparison of speech recognition algorithms. *Proceedings ICASSP*.
- Gillick, L., Ito, Y., Young, J. (1997). A probabilistic approach to confidence estimation and evaluation. *Proceedings ICASSP*.
- Godfrey, J. J., Holliman, E. C., McDaniel, J. (1992). Switchboard: Telephone speech corpus for research and development. *Proceedings ICASSP*.
- Goel, V., Kumar, S., Byrne, W. (2004). Segmental minimum Bayes-risk decoding for automatic speech recognition. *IEEE Transactions on Speech and Audio Processing* 12(3) pp. 234–249.
- Gopalakrishnan, P., Kanevsky, D., Nadas, A., Nahamoo, D. (1989). A generalization of the Baum algorithm to rational objective functions. *Proceedings ICASSP*.
- Gotoh, Y., Renals, S. (2000). Topic-based mixture language modelling. *Journal of Natural Language Engineering* 5(1) pp. 355–375.
- Gunawardana, A. (2001). CLSP research note no. 40 : Maximum mutual information estimation of acoustic HMM emission densities. Technical report, CLSP, Johns Hopkins University.
- Gunawardana, A., Byrne, W. (2001). Discriminative speaker adaptation with conditional maximum likelihood linear regression. *Proceedings Eurospeech*.
- Hain, T., Burget, L., Dines, J., Garau, G., Karafiat, M., Lincoln, M., McCowan, I., Moore, D., Wan, V., Ordelman, R., Renals, S. (2005a). The 2005 AMI system for the transcription of speech in meetings . *Proceedings MLMI*.
- Hain, T., Burget, L., Dines, J., McCowan, I., Garau, G., Karafiat, M., Lincoln, M., Moore, D., Wan, V., Ordelman, R., Renals, S. (2005b). The development of the AMI system for the transcription of speech in meetings . *Proceedings MLMI*.

- He, X., Wu, C. (2003). Minimum classification error linear regression for acoustic model adaptation of continuous density HMMs. *Proceedings ICASSP*.
- Hermansky, H. (1990). Perceptual linear predictive (PLP) analysis of speech. *Journal of the Acoustical Society of America* 87(4) pp. 1738–1752.
- Huang, X., Acero, A., Hon, H. (2001). *Spoken Language Processing: A Guide to Theory, Algorithm and System Development*. Prentice Hall.
- Itakura, F., Saito, S. (1970). A statistical method for estimation of speech spectral density and formant frequencies. *Electronics and Communications in Japan* 53A pp. 36–43.
- Janin, A., Baron, D., Edwards, J., Ellis, D., Gelbart, D., Morgan, N., Peskin, B., Pfau, T., Shriberg, E., Stolcke, A., Wooters, C. (2003). The ICSI meeting corpus. *Proceedings ICASSP*.
- Jebara, T. (2002). Discriminative, generative and imitative learning. Ph.D. thesis, Massachusetts Institute of Technology.
- Jelinek, F. (1976). Continuous speech recognition by statistical methods. *Proceedings IEEE*.
- Jiang, H., Li, X., Liu, C. (2006). Large margin hidden Markov models for speech recognition. *IEEE Transactions on Speech and Audio Processing* 14(5) pp. 1584–1595.
- Juang, B.-H., Katagiri, S. (1992). Discriminative learning for minimum error classification. *IEEE Transactions on Speech and Audio Processing* 40(12) pp. 3043–3054.
- Kaiser, J., Horvat, B., Kacic, Z. (2002). Overall risk criterion estimation of hidden Markov model parameters. *Speech Communication* 38(3-4) pp. 383–398.
- Kanevsky, D. (2004). Extended Baum transformations for general functions. *Proceedings ICASSP*.
- Kemp, T., Schaaf, T. (1997). Estimating confidence using word lattices. *Proceedings Eurospeech*.
- Kuhn, R., De Mori, R. (1990). A cache-based natural language model for speech recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 12(6) pp. 570–583.
- Kuhn, R., Junqua, J., Nguyen, P., Niedzielski, N. (2000). Rapid speaker adaptation in eigenvoice space. *IEEE Transactions on Speech and Audio Processing* 8 pp. 695–707.
- Lau, R., Rosenfeld, R., Roukos, S. (1993). Trigger-based language models: A maximum entropy approach. *Proceedings ICASSP*.
- Lee, L., Rose, R. (1996). Speaker normalisation using efficient frequency warping procedures. *Proceedings ICASSP*.

- Leggetter, C. J., Woodland, P. (1995). Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models. *Computer Speech and Language* 9(2) pp. 171–185.
- Leonard, R. G. (1984). A database for speaker-independent digit recognition. *Proceedings ICASSP*.
- Li, J., Yuan, M., Lee, C.-H. (2006). Soft margin estimation of hidden Markov model parameters. *Proceedings Interspeech*.
- Li, X., Jiang, H., Lui, C. (2005). Large margin HMMs for speech recognition. *Proceedings ICASSP*.
- Lippmann, R. P. (1997). Speech recognition by machines and humans. *Speech Communication* 22(1) pp. 1–15.
- Lui, C., Jiang, H., Li, X. (2005). Discriminative training of CDHMMs for maximum relative separation margin. *Proceedings ICASSP*.
- Macherey, W., Haferkamp, L., Schluter, R., Ney, H. (2005). Investigations on error minimizing training criteria for discriminative training in automatic speech recognition. *Proceedings Interspeech*.
- MacKay, D. (1991). Bayesian methods for adaptive models. Ph.D. thesis, Caltech.
- Mandal, A., Ostendorf, M., Stolcke, A. (2006). Speaker clustered regression-class trees for MLLR. *Proceedings Interspeech*.
- Mangu, L., Brill, E., Stolcke, A. (1999). Finding consensus among words: Lattice-based word error minimization. *Proceedings Eurospeech*.
- Manning, C. D., Schütze, H. (June 1999). Foundations of Statistical Natural Language Processing. MIT Press.
- Martin, A., Doddington, G., Kamm, T., Ordowski, M., Przybocki, M. (1997). The DET curve in assessment of detection task performance. *Proceedings Eurospeech*.
- McDermott, E. (1997). Discriminative training for speech recognition. Ph.D. thesis, Waseda University.
- McNemar, I. (1947). Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika* 12 pp. 153–157.
- Na, K., Jeon, B., Chang, D., Chae, S., Ann, S. (1995). Discriminative training of hidden Markov models using overall risk criterion and reduced gradient descent method. *Proceedings Eurospeech*.

- Nadas, A., Nahamoo, D., Picheny, M. (1988). On a model-robust training algorithm for speech recognition. *IEEE Transactions on Acoustics, Speech and Signal Processing* 36 pp. 1432–1435.
- Normandin, Y. (1991). Hidden Markov models, maximum mutual information estimation and the speech recognition problem. Ph.D. thesis, McGill University.
- Oppenheim, A. V., Schafer, R. W., Stockham, A. G. (1968). Nonlinear filtering of multiplied and convolved signals. *Proceedings of the IEEE* 56(8) pp. 1264–1291.
- Padmanabhan, M., Ramabhadran, B., Eide, E., Ramaswamy, G., Bahl, L., Gopalakrishnan, P., Roukos, S. (1997). Transcription of new speaking styles - Voicemail. *Proceedings DARPA Hub4 Workshop*.
- Pallett, D. (2003). A look at NIST’s benchmark ASR tests: past, present, and future. *Proceedings ASRU*.
- Pallett, D. S., Fisher, W. M., Fiscus, J. (1990). Tools for the analysis of benchmark speech recognition tests. *Proceedings ICASSP*.
- Pitz, M. (2005). Investigations on linear transformations for speaker adaptation and normalization. Ph.D. thesis, Aachen University.
- Pitz, M., Wessel, F., Ney, H. (2000). Improved MLLR speaker adaptation using confidence measures for conversational speech recognition. *Proceedings ICSLP*.
- Povey, D. (2003). Discriminative training for large vocabulary speech recognition. Ph.D. thesis, Cambridge University.
- Povey, D., Gales, M., Kim, D. Y., Woodland, P. (2003a). MMI-MAP and MPE-MAP for acoustic model adaptation. *Proceedings Eurospeech*.
- Povey, D., Woodland, P. (2002). Minimum phone error and I-smoothing for improved discriminative training. *Proceedings ICASSP*.
- Povey, D., Woodland, P., Gales, M. (2003b). Discriminative MAP for acoustic model adaptation. *Proceedings ICASSP*.
- Price, P., Fisher, W., Bernstein, J., Pallett, D. (1988). A database for continuous speech recognition in a 1000-word domain. *Proceedings ICASSP*.
- Reichl, W., Ruske, G. (1995). Discriminative training for continuous speech recognition. *Proceedings Eurospeech*.
- Schluter, R., Macherey, W., Muller, B., Ney, H. (2001). Comparison of discriminative training criteria and optimization methods for speech recognition. *Speech Communication* 34 pp. 287–310.

- Schluter, R., Muller, B., Wessel, F., Ney, H. (1999). Interdependence of language models and discriminative training. *Proceedings IEEE ASRU Workshop*.
- Shinoda, K., Lee, C. (2001). A structural Bayes approach to speaker adaptation. *IEEE Transactions on Speech and Audio Processing* 9(3) pp. 276–287.
- Steiger, J. (1980). Tests for comparing elements of a correlation matrix. *Psychological Bulletin* 87 pp. 245–251.
- Stevens, S. S., Volkman, J., Newman, E. B. (1937). A scale for the measurement of the psychological magnitude pitch. *Journal of the Acoustical Society of America* 8(3) pp. 185–190.
- Tsakalidis, S., Doumptiotis, V., Byrne, W. (2002). Discriminative linear transforms for feature normalization and speaker adaptation in HMM estimation. *Proceedings ICSLP*.
- Valtchev, V., Odell, J., Woodland, P., Young, S. (1997). Mmle training of large vocabulary recognition systems. *Speech Communication* 22(4) pp. 303–314.
- Vapnik, V. N. (1998). *Statistical Learning Theory*. John Wiley & Sons.
- Venkataramani, V., Chakrabartty, S., Byrne, W. (2007). Gini support vector machines for segmental minimum Bayes risk decoding of continuous speech. *Computer Speech and Language* 21(3) pp. 423–442.
- Viterbi, A. J. (1967). Error bounds for convolutional codes and an asymptotically optimal decoding algorithm. *IEEE Transactions on Information Theory* 13(2) pp. 260–269.
- Wang, L., Woodland, P. (2002). Discriminative adaptive training using the MPE criterion. *Proceedings ASRU*.
- Wang, L., Woodland, P. (2004). MPE-based discriminative linear transform for speaker adaptation. *Proceedings ICASSP*.
- Wang, L., Woodland, P. (2008). MPE-based discriminative linear transforms for speaker adaptation. *Computer Speech and Language* 22(3) pp. 256–272.
- Watanabe, S., Minami, Y., Nakamura, A., Ueda, N. (2003). Application of variational Bayesian approach to speech recognition. *Advances in Neural Information Processing Systems* 15.
- Weinstein, C. J. (1991). Opportunities for advanced speech processing in military computer-based systems. *Proceedings of the IEEE* 79(11) pp. 1626–1641.
- Wessel, F., Schluter, R., Macherey, K., Ney, H. (2001). Confidence measures for large vocabulary continuous speech recognition. *IEEE Transactions on Speech and Audio Processing* 9(3) pp. 288–298.

- Williams, J., Young, S. (2007). Partially observable Markov decision processes for spoken dialog systems. *Computer Speech and Language* 21 pp. 393–442.
- Woodland, P. (2001). Speaker adaptation for continuous density HMMs: A review. *Proceedings ITRW Adaptation Methods for Speech Recognition*.
- Woodland, P., Gales, M., Pye, D., Young, S. (1997). Broadcast news transcription using HTK. *Proceedings ICASSP*.
- Woodland, P., Povey, D. (2002). Large scale discriminative training of hidden Markov models for speech recognition. *Computer Speech and Language* 16(1) pp. 25–37.
- Wu, J., Huo, Q. (2002). Supervised adaptation of MCE-trained CDHMMS using minimum classification error linear regression. *Proceedings ICASSP*.
- Young, S., Everman, G., Hain, T., Kershaw, D., Moore, G., Odell, J., Ollason, D., Povey, D., Valtchev, V., Woodland, P. (2003). The HTK book for HTK version 3.2.1.
- Young, S., Odell, J., Woodland, P. (1994). Tree-based state tying for high accuracy acoustic modelling. *Proceedings ARPA Workshop on Human Language Technology*.
- Yu, D., Deng, L., He, X., Acero, A. (2008). Large-margin minimum classification error training: A theoretical risk minimization perspective. *Computer Speech and Language* 22(4) pp. 415–429.
- Yu, K., Gales, M. (2005). Bayesian adaptation and adaptively trained systems. *Proceedings ASRU*.
- Yu, K., Gales, M. J. F. (2006). Discriminative cluster adaptive training. *IEEE Transactions on Audio, Speech and Language Processing* 14(5) pp. 1694–1703.
- Zheng, J., Stolcke, A. (2005). Improved discriminative training using phone lattices. *Proceedings Interspeech*.
- Zu, Y.-Q. (1997). Sentences design for speech synthesis and speech recognition database by phonetic rules. *Proceedings Eurospeech*.
- Zu, Y.-Q., Li, W.-X., Ho, M.-C., Chan, C. (1996). HKU96-A Putonghua corpus. Technical report, University of Hong Kong, China.
- Zwicker, E. (1961). Subdivision of the audible frequency range into critical bands. *Journal of the Acoustical Society of America* 33(2) pp. 248–249.